# A markovian model for stochastic integrate-and-fire networks

Jonathan Touboul

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *A markovian model for stochastic integrate-and-fire networks*

Jonathan Touboul

**N° 6661**

September 22, 2008

Thème BIO


ISSN 0249-6399 ISRN INRIA/RR--6661--FR+ENG

INRIA
SOPHIA ANTIPOLIS

# A markovian model for stochastic integrate-and-fire networks

Jonathan Touboul *

Thème BIO — Systèmes biologiques
Projet Odyssée

Rapport de recherche n° 6661 — September 22, 2008 — 42 pages

**Abstract:** In this paper we introduce and study a mathematical framework in order to characterize and simulate networks of noisy integrate-and-fire neurons. This framework is based on a markovian modelization of the network, similar to the event-based modelization of deterministic networks. In these networks the value of interest at each neuron is not the membrane potential itself but the related *countdown process*, which is defined loosely as the time remaining to the next spike if nothing occurs meanwhile in the network. The main issue of this modelization is to ensure that the dynamics of this countdown process, possibly supplemented with other variables, is an autonomous Markov process (i.e. that does not depend on the membrane's potential).

We prove that a wide range of integrate-and-fire neuron models and different types of interactions fit into this general mathematical framework. This framework involves renewal processes and has already been studied in the field of random networks in a more restricted setting by Cottrell, Robert, Turova for instance [6, 7, 13, 27, 28], and from a mathematical viewpoint, ergodicity matters have been discussed Fayolle, Menshikov, Malyshev and Borovkov [12, 3].

This modelization provides a very efficient algorithm to simulate large networks of noisy integrate-and-fire neuron models. We discuss different types of implementations, and developped together with Renaud Kervien and Alexandre Chariot a very efficient paralel simulator implement on GPU.

**Key-words:**

* jonathan.touboul@sophia.inria.fr

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

# A markovian model for stochastic integrate-and-fire networks

**Résumé :** In this paper we introduce and study a mathematical framework in order to characterize and simulate networks of noisy integrate-and-fire neurons. This framework is based on a markovian modelization of the network, similar to the event-based modelization of deterministic networks. In these networks the value of interest at each neuron is not the membrane potential itself but the related *countdown process*, which is defined loosely as the time remaining to the next spike if nothing occurs meanwhile in the network. The main issue of this modelization is to ensure that the dynamics of this countdown process, possibly supplemented with other variables, is an autonomous Markov process (i.e. that does not depend on the membrane's potential).

We prove that a wide range of integrate-and-fire neuron models and different types of interactions fit into this general mathematical framework. This framework involves renewal processes and has already been studied in the field of random networks in a more restricted setting by Cottrell, Robert, Turova for instance [6, 7, 13, 27, 28], and from a mathematical viewpoint, ergodicity matters have been discussed Fayolle, Menshikov, Malyshev and Borovkov [12, 3].

This modelization provides a very efficient algorithm to simulate large networks of noisy integrate-and-fire neuron models. We discuss different types of implementations, and developped together with Renaud Kervien and Alexandre Chariot a very efficient paralel simulator implement on GPU.

**Mots-clés :**

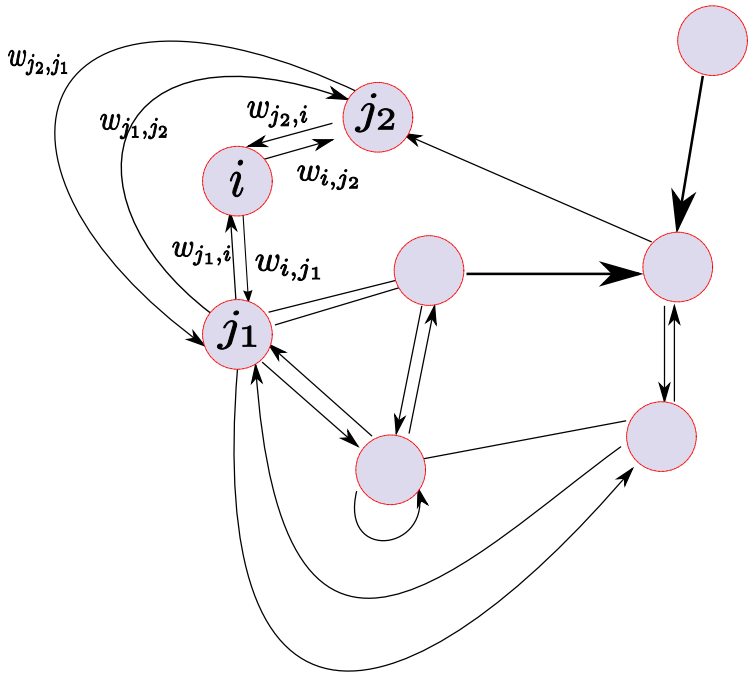



Figure 1: A general neural network architecture: the network is composed of neurons (blue circles) connected through a directional connectivity map (black arrow) with synaptic efficiency $w_{ij}$. The intrinsic dynamics and the effect of an incoming spike on the postsynaptic neuron can be modeled in many ways

# 1 Theoretical framework

In this paper we build a bridge between a wide range of biological networks models and a general mathematical framework. The type of network we consider is composed of $N$ stochastic integrate-and-fire neurons(see figure 1). Classicaly, neuron's activity is described by its membrane potential. The membrane potential's dynamics we consider in this paper is stochastic: each neuron receives at his synapses noisy inputs corresponding to the random activity of ion channels and at the external activity of the network, as reviewed [24]. This random spike incoming is here modelled as Brownian motion, using a diffusion approximation. Different types of intrinsic dynamics and of synaptic integration will be considered ans can coexist in a given network.

During the time intervals where no spike is emitted in the network, the membrane potential of each neuron evolves as independently to the other's, according to its intrinsic dynamics. When the membrane potential $V^{(i)}(t)$ of the neuron indexed by $i$ reaches its

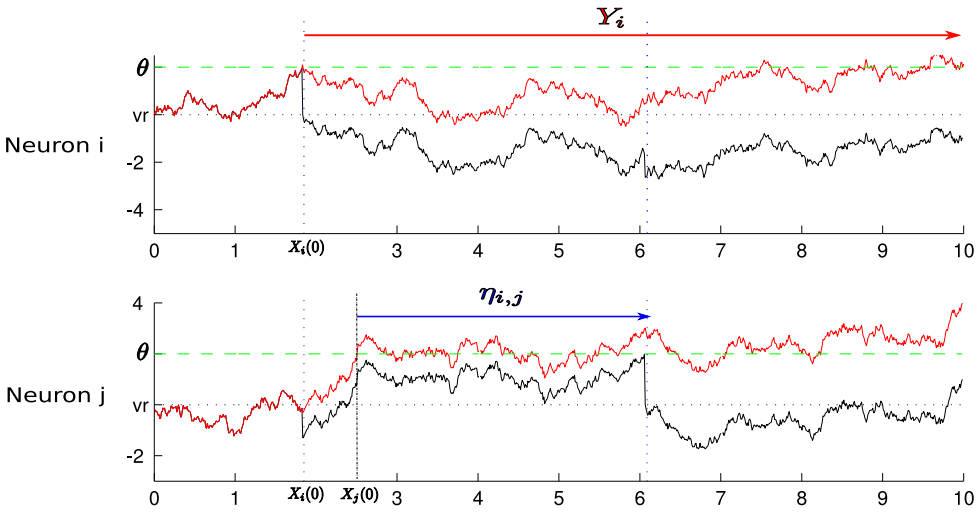



Figure 2: A sample trace of the membrane potential for two connected neurons index by $i$ and $j$. The neuron $i$ is the first to spike in the network: it has the lowest first spike time $X_i(0)$. At this time, the neuron $i$ is reset to $v_r$ and its next spike time is reset according to $Y_i$. It sends a spike to its neighbors, among which $j$. If the interaction is instantaneous, the membrane potential of $j$ is instantaneously added the synaptic weight $w_{i,j}$ and the time to the next spike for the neuron $j$ is increased by a value $\eta_{i,j}$. This figure was produced in the case of the Perfect Integrate-and-fire model.

deterministic threshold function $\theta(t)$ at time $t_0$, the neuron elicits an action potential. Subsequently, its membrane potential is reset to a given value $V_r^{(i)}$, and the states of all the postsynaptic neurons $j$ connected to the neuron $i$ is modified. We denote by $\mathcal{V}(i)$ the postsynaptic neighboorhood of the neuron $i$, i.e. the set of neurons that receive spikes from neuron $i$. The effect of a presynaptic spike received by neuron $j \in \mathcal{V}(i)$ can be modelled in different fashions: it can be considered as having an instantaneous effect on the membrane potential (i.e. $V^{(j)}(t_0) = V^{(j)}(t_0^-) + w_{i,j}$ where $w_{i,j}$ is the synaptic efficiency of the connection $i \to j$), or more complex, including for instance a synaptic current, a synaptic pulses, etc.... Many examples will treated in the text. Figure 2 illustrates the dynamics of the network, showing the structure of the network in 1 and the dynamics of the membrane potential in 2

This type of model was studied for instance by Brunel and Hakim [5] with the use of the Fokker-Planck equation. Assuming that the network is sparsely connected, they found that in the limit $N \to \infty$ the network exhibited a sharp transition between two regimes: a stationnary regime and a weakly synchronized oscillatory regime. In their model, each neuron is an integrate-and-fire neuron, and is randomly connected to C neurons of the

network, and to $C_{\text{ext}}$ external neurons. The sparse connectivity assumption is $\varepsilon = \frac{C}{N} \ll 1$. Interactions between external and internal neurons are delayed by a constant delay $\delta$ (i.e. when a spike is emitted by a neuron of the network, it decreases or increases the membrane potential after a time $\delta$, see section 4). This delay plays a crucial role in the generation of global oscillations. We wish to re-express the dynamics from an event-driven point of view (see for example Reutimann, Giugliano an Fusi [20]), and to consider the noise in the dynamics of each individual neuron.

Independently, in the field of stochastic networks and queue theory and Markov processes, a network model has been developed during the last 10 years. It is referred for instance as the *hourglass model* by Turova [2, 7, 28, 27]. This model has been introduced for the first time by Marie Cottrell in [6], and the variable taken into account was initially called the *inhibition process*. This name is very confusing in the field of neuroscience, and we will not use this expression when dealing with neuron and prefer the name of *countdown process*.

These models are defined by two random parameters:

- The random variables $(Y_i)_{i=1\dots N}$ which describe the interspike interval distribution for the neuron $i$.
- $(\eta_{i,j})_{i\neq j}$ describing the interaction of $i \to j$.

Let the state of the network be described by a N-dimensional vector $(X_t)_{t\geq 0} = \left((X_t^{(i)})_{i=1\dots N}\right)_{t\geq 0}$ having the following dynamics: let $t > 0$,

(i). if $\forall i \in \{1 \dots N\}$, $X_i(t) > 0$ then each component of $X$ decreases linearly with slope $-1$ in time.

(ii). if $\exists i \in \{1 \dots N\}$, $X_i(t^-) = 0$, subsequently we have:

- $X_i$ is reset to a random variable independent of all the history of the process and with distribution $Y_i$.
- $\forall j \in \mathcal{V}(i)$, a positive random variable $\eta_{i,j}$ is added to $X_j$:

$$X_j(t) = X_j(t^-) + \eta_{i,j}$$

Hence each node of the network is a renewal process and the network structure makes these processes interact via positive random variables.

In this paper we build a bridge between these two models. We will see that stochastic networks of integrate-and-fire neurons can be described using an extension of the hourglass model, but need a more general formalism to take into account the more complex interaction structure at the level of the membrane potential.

## 2 From Biological networks to the Hourglass model

In this section we present the equivalence between the bio-inspired network and the hourglass model. We first consider inhibitory networks for the sake of simplicity. Indeed, in the case of non-inhibitory networks can appear the phenomenon of what we call a *spike avalanche*. Assume that the interactions are considered instantaneous and excitatory. In this case the following process can occur: if the synaptic efficiencies are big enough, the spike emission of a neuron can induce at the very same time the spike emission of the neurons directly connected to this neuron, which themselves can induce spikes in their neighboorhood. A spike can therefore be transmitted in the whole network, and then induce the spike emission in the first neuron who spike, and therefore this process will not stop. This mechanism is clearly not biologically plausible: first of all there are transmission delays in the network, and hence this avalance, even if it occurs, generates a high frequency activity, but with no logical problems such as the one we just described. Furthermore, the limited resources in the neuron's environment makes such a wasteful enegetic process impossible. From a comutational and theoretical point of view, such a phenomenon results in stucking the dynamics at the time when it occurs: this infinite loop of simultaneous spikes blocks the process at this time and we cannot infer what would happen afterwards.

We will see in section 4 that including a refractory period and transmission delays between neurons overcomes this difficulty.

This equivalence is built upon the introduction of a new process related to the membrane potential process, the *countdown process*, rigorously defined as follows:

**Definition 2.1.** *[Countdown process] For each neuron $i$, let us define $X^{(i)}(t) \geq 0$ the duration of time (after time $t$) till the first firing moment of this neuron, if no interaction takes place meanwhile. We will call this stochastic process the countdown process of the neurons.*

This process is called *countdown* because of its dynamics, but in fact at any time, its value gives us the time to wait till the next spike, so it can be also seen as a *clock*. It can be seen as a countdown set at the instant of reception of the last spike or just after the spike, to the time to wait for the next spike to occur if no interaction takes place meanwhile. The dynamics of this variable $X^i$ is linearly decreasing with slope $-1$ during the intervals of time where no spike is received or produced:

$$\frac{dX^{(i)}}{dt} = -1 \tag{2.1}$$

At time $t$, the next spike will occur in neuron $i = \operatorname{Arg\,Min}_{j\in 1\dots N} X^{(j)}(t)$ at time $t + X^{(i)}(t)$ ($t$ is the absolute time). In most of the case, for instance in the case where all the random variables have a density with respect to Lebesgue's measure, the probability for two neurons to spike exactly at the same time is null when the network is inhibitory. In that case, we neglect this case and assume that only one neuron spikes at a given time. At spike time, $X^{(i)}(t)$ is instantaneously reset by drawing the law of a random variable noted $Y_i$, which has

the same distribution as the first hitting time of the stochastic process $(V_t^{(i)})_{t\geq 0}$ starting from $V_r$ to the boundary $\theta(t)$ (the distribution of the interspike interval in terms of neural models). The states of all neurons just before the spike are given by: $X^{(j)}\left((t+X^{(i)})^-\right) = X^{(j)}(t) - X^{(i)}(t)$. Finally, the states of all neurons $j$ connected to neuron $i$ are modified according to the spike produced by neuron $i$. Because the interaction is inhibitory, this amounts to postponing the spike produced by neuron $j$ by an amount $\eta_{i,j} \geq 0$ (see Fig 3), because the inhibition increases the time to the next spike.

In general, $\eta_{i,j}$ is a random variable depending on the membrane potential $V^{(j)}$ at time $t$. In most of the models considered in section 3, it depends in fact only on $X^{(j)}$, so that the update reads $X^{(j)}(t+X^{(i)}) = X^{(j)}(t) - X^{(i)}(t) + \eta_{i,j}(X^{(j)}(t) - X^{(i)}(t))$, where $\eta_{i,j}(x)$ is a random function.

In all our mathematical study we consider the process

$$X(t) := (X^{(i)}(t))_{1\leq i\leq N} \tag{2.2}$$

Up to an additional Markov chain, this model will be a continuous time Markov process, as we will show in section 3. The process $(X_t)_t$ defined is piecewise continuous, so the analysis of Davis in [11] can be applied here. Our case is even more simple since the discontinuities are very simply related to the value of the process. This very particular property implies that studying the continuous time stochastic process is strictly equivalent to considering one of the two folloving discrete time Markov chain (2.3) and (2.4).

Indeed, let $(t_n)$ denote the time sequence of the spikes emitted by one of all the neurons, $(Z_n)$ the sequence of the states just before each spike and $(X_n)$ the vector of states just after each spike.

$$Z_n = X(t_n^-) \tag{2.3}$$

$$X_n = X(t_n) \tag{2.4}$$

Consider now the random variable $\eta_{i,j}$ to add to the state of a postsynaptic neuron $j$ when receiving a spike from $i$ at time $t^*$. This random variable is the delay caused by the inhibition, i.e. the additional time to wait for $j$ to spike because of the reception of a presynaptic spike.

All the work done in the following sections 3 and 4 is aimed to show that many biological neuron models fit into the framework described in section 2 and to identify the parameters of the corresponding Hourglass model. We will see that in many cases, these random variables can be related to first hitting times of stochastic processes.

# 3 Inhibitory Networks with instantaneous interactions

In this section we consider different types of models of linear integrate-and-fire neurons and different types of inhibitory synaptic interactions, and up to a transformation show that the network model can be considered as an hourglass network, and identify the parameters of the model.

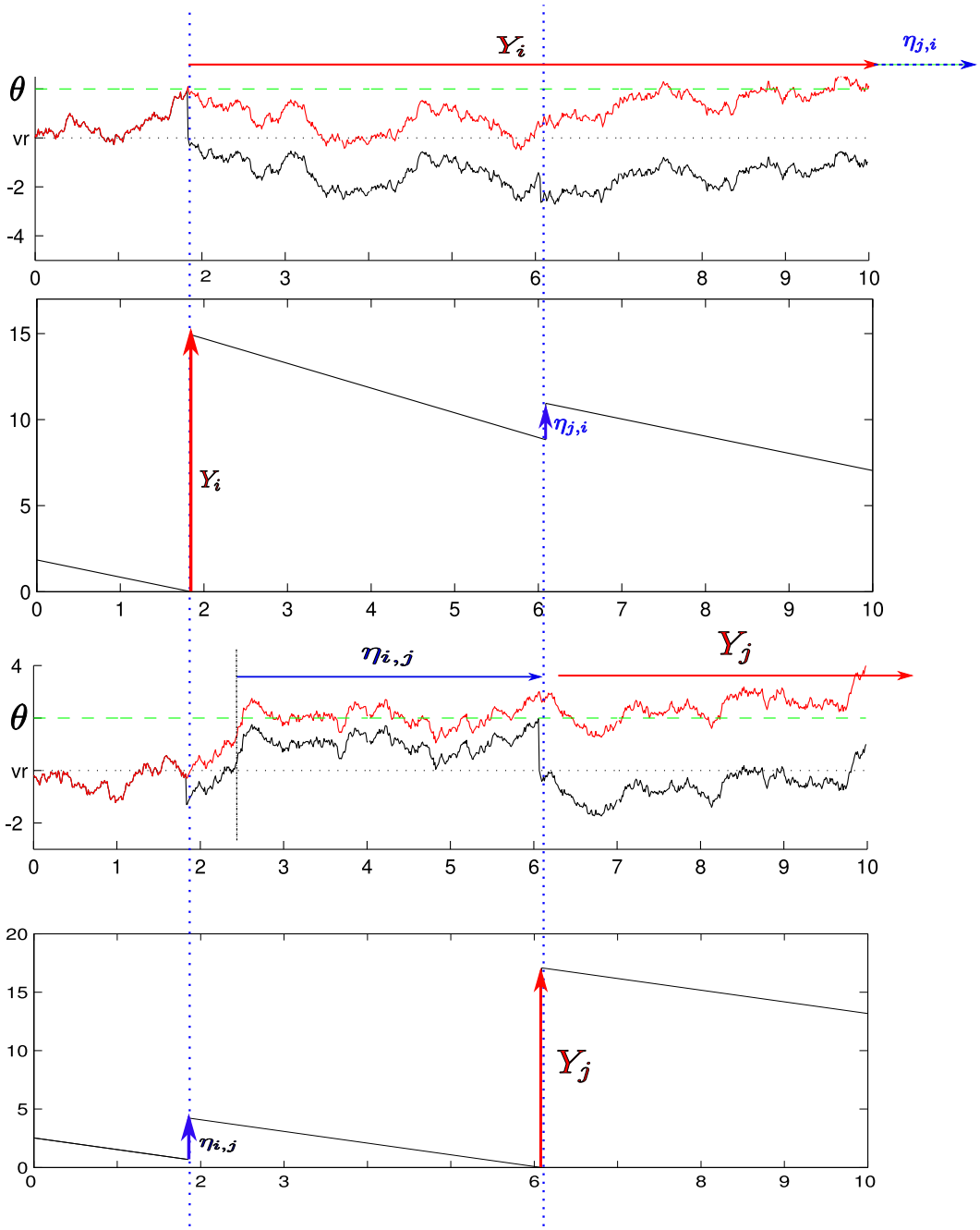



Figure 3: A representation of a sample path for the countdown process and the related membrane potential in the case of the perfect integrate-and-fire neuron represented in figure 2.

The first model we consider is a noisy integrate-and-fire neuron without leak current, which we refer as the perfect noisy integrate-and-fire neuron. We then add a leak current.

We first state some general results about these random variables. First of all, it is very interesting to note that the reset process is only linked with the presynaptic neuron, and has the law of the first hitting time of the membrane potential process to the threshold function $\theta(t)$. Indeed, when a neuron elicits a spike in the integrate-and-fire framework with no refractory period, its membrane potential is reset to a certain value $V_r$[1] Therefore the reset random variable, defined as the time before the next spike of the neuron, has the law of the first hitting time of the membrane potential to the threshold. The interaction variable only depends on the postsynaptic dynamics of the membrane potential and on the synaptic efficiency $w$. When the neuron $j$ receives a spike from the neuron $i$ at time $t$, the time to the next spike is changed, and the random variable corresponding is equal to the difference between the time to reach the threshold starting from $V^{(j)}(t) + w_{ij}$ and the time to reach the threshold starting from $V^{(j)}(t)$. Hence in the general case, this random variable depends on the value of the potential at time $t$. We will see that in the simplest cases we treat here this random variable only depends on $X^{(j)}$, the time to the next spike for the postsynaptic neuron $j$. This property is very interesting since it makes the countdown process an independent Markov chain, i.e. that depends on no other process.

## 3.1 Perfect integrate-and-fire models

### 3.1.1 Perfect IF neuron with instantaneous synapses

We start by considering the perfect integrate-and-fire neuron with external inputs and Brownian noise. The membrane potential of the neuron $i$, denoted $V^{(i)}(t)$, is hence driven by the following equation between two spikes:

$$\tau_i dV^{(i)}(t) = I_e^{(i)}(t)dt + \sigma_i dW_t^{(i)} \tag{3.1}$$

where $\tau_i$ is the membrane potential time constant, $I_e^{(i)}(t)$ is the input current, $\sigma_i$ the standard deviation of the noise and $(W^{(i)})_{1\leq i\leq N}$ are independent Brownian motions, which represents external synaptic stimulations[2]. The neuron fires when its membrane potential reaches the threshold $\theta$: the membrane potential is reset to a value $V_r$ and a spike is emitted.

$$V^{(i)}(t^-) = \theta \Rightarrow V^{(i)}(t) = V_r \tag{3.2}$$

[1] The reset value $V_r$ can also be a random variable with no additional complexity. The results we obtain for a constant reset value can be readily extended to this more general model.

[2] It could have been possible to replace the Brownian motions by instantaneous spikes ($V^{(i)} \to V^{(i)} + \delta$) triggered according to a Poisson process (the equation (3.1) would be the diffusion approximation of this type of excitation). This would change considerably the following study, since the process is no more continuous between two consecutive spikes

In the absence of interactions, $V^{(i)}(t)$ integrates the entry $I_e^{(i)}$ with an additive noise proportional to a Brownian motion, i.e. :

$$V^{(i)}(t) = \int_0^t I_e^{(i)}(s)\, ds + \sigma_i W_t^{(i)} \tag{3.3}$$

In this model, we also consider instantaneous inhibitory synaptic interactions between neurons. More precisely, when the neuron $j$ receives a spikes from a presynaptic neuron $i$ spikes at time $t^*$, then its membrane potential is instantaneously added the synaptic weight:

$$\forall j \in \mathcal{V}(i) V^{(j)}(t^*) = V^{(j)}(t^{*-}) + \omega_{i,j} \tag{3.4}$$

The related countdown process $X^{(i)}$ is defined by the interaction random variable and the reset random variable $\tau$ defined by:

$$\tau := \inf\left\{ t > 0;\ \ W_t^{(i)} = \frac{1}{\sigma_i}(\theta(t) - \int_0^t I_e(i)(s)\, ds) \right\} \tag{3.5}$$

This random variable is hence the first hitting time of the Brownian motion to a curved boundary. Since the Brownian motion is a Gauss-Markov process, this law can be computed by Volterra's, Durbin's or Girsanov's method, as reviewed in [24]. In the case where the input current and the threshold function are constant, closed form expressions of the pdf of this law are provided using martingales methods (or the reflection principle) together with Girsanov's theorem.

The interaction random variable is deduced by the effect of a presynaptic spike incoming at a synapse. When we consider fully instantaneous synapses, i.e. integrating the noise as a Brownian motion and the interactions as Dirac functions: when the spike is emitted, the postsynaptic neuron's membrane potential is instantaneously changed. Therefore, when the neuron $j$ receives an inhibitory spike from neuron $i$ at time $t$, the time of the next spike of neuron $j$ is $t + X^{(j)}(t) + \eta_{i,j}$, where $\eta_{i,j}$ is the first hitting time of the drifted Brownian motion to the boundary $\theta$, starting from $\theta + w_{i,j}$ (recall that in that case, $w_{i,j} \leq 0$). Since the stochastic process solution of (3.1) is a Lévy process, this random variable is the first hitting time of a drifted Brownian motion starting from 0 to the constant barrier $w_{i,j}$.

The density of this random variable reads:

$$p^{(j)}(t) = \frac{|w_{i,j}|}{\sqrt{2\pi t^3}} e^{-\frac{(w_{i,j} - \mu_j\, t)^2}{2t}} \mathbb{1}_{\mathbb{R}_+^*}(t) \tag{3.6}$$

Thus in the case of the perfect integrate-and-fire model, the effect of the reception of a spike is equivalent to adding an independent random variable with the probability density w.r.t. Lebesgue's measure given by (3.6).

In the case of stationary inputs, we can see that the countdown process is an autonomous Markov process. In the case of a time-varying input, the process $(X_t, t)$ is an autonomous

Markov process. Furthermore, we can readily prove that the times of the spikes of these chains have the same probability distribution as the times of the spikes computed using the membrane potential representation.

### 3.1.2 Perfect integrate-and-fire neuron with synaptic integration

If we still consider that the time constant of the leak is very large compared to the time scale of the observation, of the inputs and compared to the firing rate, and furthermore that the noisy interactions are integrated with a time constant $\tau_s \neq 0$, then we obtain the following equations for the membrane potential:

$$\begin{cases} \frac{\mathrm{d}V_t}{\mathrm{d}t} & = I_e(t) + I_s(t) \\ \tau_s dI_s(t) & = -I_s(t)dt + \sigma_s dW_t \end{cases} \tag{3.7}$$

whose solution read:

$$V_t = V(0) + \int_0^t I_e(s)\,ds + \tau_s(1 - e^{-t/\tau_s})I_s(0) + \sigma \int_0^t \int_0^s e^{-(s-u)/\tau_s}\,dW_u\,ds,$$

expression that involves a DIP.

If we further consider that $\tau_s$ is very big compared to the time constants of the experience, we obtain the perfect integrate-and-fire model with perfect synaptic currents:

$$V_t = V(0) + \int_0^t I_e(s)\,ds + I_s(0)t + \sigma \int_0^t W_s\,ds,$$

which involves an IWP.

The reset random varibale in the case of the exponentially decaying synaptic conductances has the law the first hitting time of the DIP $\int_0^t \int_0^s e^{-(s-u)/\tau_s}\,dW_u\,ds$ to a curved boundary depending on the inputs of the neuron and the initial condition of the synaptic input. This hitting time can be approximated using the framework we developped in [25] and no closed-form solution can be provided.

In the case where the decay time of the synapse is not taken into account, this reset random variable has the law of the first hitting time of an IWP to the curve $V(0) + \int_0^t I_e(s)\,ds + I_s(0)t$. Therefore, we have closed-form expressions for the pdf of the law of this random variable for polynomials input currents of order lower or equal to 2 (see [25]), depending on the initial condition on the input current $I_s(0)$.

Using the linearity of the equation, we can compute the interaction variable. In the case of instantaneous synaptic integration of the integration, this random variable has the law of the first hitting time of the threshold $\theta$ of the membrane potential process starting from $(\theta + w_{ij}, I_s(X_j))$ to reach the threshold $\theta$, and can therefore be computed using the same approximations or formulas depending on the model we choose and the type of input current considered.

The case of integrated inputs is more logical: it assumes that everything coming through the synapse is integrated following the same differential equation. In that case the effect of an

incoming spike on a postsynaptic neuron is added instantaneously to the synaptic current. Therefore, using the same technique as before, we can obtain the law of the interaction variable. For the perfect integrate and fire neuron with exponentially decaying synaptic current, the law of this random variable is deduced from the law of the first hitting time of the related DIP starting from $(\theta + w_{ij}\tau(1 - e^{-t/\tau_s}), I_s(X_j) + w_{ij}e^{-X_j/\tau})$ to reach the threshold $\theta$. In the case of the perfect integrate-and-fire neuron with perfect synapses, the law of the interaction random variable is given by the law of the first hitting time of the related IWP starting from $(\theta + tw_{ij}, I_s(X_j) + w_{ij})$ to reach the threshold $\theta$.

In these cases, we observe that the coutdown process is no more a Markov chain. Indeed, in order to compute the reset random variable, we need to consider the value of the synaptic current at the spike time (we will see that this variable can be computed). Furthermore, for non stationary inputs, we have to add the time as a new variable. Considering the countdown process augmented of the value of the synaptic current at the time of the next spike $I_s^{(n)}$ and of the time $t$ is a Markov process whose spike times (times where a coordinate of the countdown process is 0) have the same probability distribution as the spikes in the network. Eventually, this chain can be considered as a discrete time Markov chain if we sample it at the times of the spike. Let us precise the dynamics of this process. Consider that this process after the $n^{\text{th}}$ spike is $(X^n, I_s^n, t^n)$. Then the next spike will be fired from the neuron $i_n$ having the lowest countdown value. It will fire at time $t^{n+1} = t^n + X_{i_n}^n$. Its countdown value will be reset to the first hitting time of the related DIP or IWP, and as we have seen in [25], the law of the pair composed of next spike time and the relative location of the synaptic currents at this time is known. Therefore by drawing in the law of this pair, we have thus the new countdown value and the future synaptic current at the time of the next spike for $i_n$. Similarly, each neuron $j \in \mathcal{V}(i)$ is updated according to the law of the first hitting time of the related membrane's potential starting with input current given by $I_j^n$ to reach a given threshold, and therefore the new countdownw value and the future location of the input current are computed at the same time using the results of [25]. The other neurons' state are unchanged. Between two spike times, the variable of synaptic current is unchanged, the time increases linearly with slope 1 and the countdown decreases linearly with slope $-1$. It is clear that the law of the spikes is the same as the law of the zeros of the countdown process.

## 3.2 Leaky integrate-and-fire models with instantaneous synapses

We now take into account the leak of the membrane potential, but still consider the synaptic integration instantaneous. The general Leaky Integrate-and-Fire (LIF) equation with instantaneous synaptic and noisy input currents reads :

$$\left\{ \begin{array}{lcl} \tau_i dV^{(i)} & = & -(V^{(i)} + I_e(t))dt + \sigma_i dW_t^{(i)} \\ V^{(i)}(t^-) & = & \theta \Rightarrow V^{(i)}(t) = V_r \end{array} \right. \tag{3.8}$$

where $(W_t^i)_{1 \leq i \leq N}$ are independent Brownian motions. The reset random variable is the same for all synaptic interactions. It is distributed as the hitting time of the threshold $\theta$

starting from $V_r$ of the process defined by (3.8). The only difficulty arises from the current input $I_e$: if it depends on the time $t$, then this random variable has not the same law at each time but depends on the time of the spike. If $I_e$ is constant, then:

$$Y_i := \inf\left\{t > 0;\ \ V_t^{(i)} = \theta | V_0^{(i)} = V_r\right\} \tag{3.9}$$

where $V^{(i)}$ is solution of (3.8). If $I_e$ is not constant, then assume that the neuron $i$ spikes at time $t^*$. At this time, the process $X^{(i)}$ is reset by drawing an independent random variable having the law of (3.9) where $V^{(i)}$ is solution of (3.8) with the time-shifted input current $I_e^{(i)'}(t) = I_e^{(i)}(t + t^*)$.

We consider that the membrane potential follows the equation (3.8), together with the spiking condition:

$$V^{(i)}(t^-) = \theta \Rightarrow \left\{ \begin{array}{rcl} V^{(i)}(t) & = & V_r \\ V^{(j)}(t) & = & V^{(j)}(t^-) + w_{i,j}\mathbb{1}_{j\in\mathcal{V}(i)} \end{array} \right. \tag{3.10}$$

We compute the membrane potential with and without the reception of a spike. Let $t^*$ be the time when the neuron $j$ receives a spike, $V^{(j)}$ the membrane potential of the neuron $j$ after reception of a spike, $\widetilde{V}^{(j)}$ the membrane potential of the neuron $j$ without any interaction with other neurons, $V^*_{(j)} := V^{(j)}(t^{*-})$ and $X^*_{(j)} := X^{(j)}(t^{*-})$. We have :

$$V^{(j)}(t^* + t) = (V^*_{(j)} + w_{i,j})e^{-t/\tau} + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau} I_e^j(s + t^*)\, ds + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau}\sigma\, dW_s$$

and

$$\widetilde{V}^{(j)}(t^* + t) = V^*_{(j)}e^{-t/\tau} + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau} I_e^j(s + t^*)\, ds + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau}\sigma\, dW_s$$

From the two equations above we can easily see that :

$$V^{(j)}(t^* + t) = \widetilde{V}^{(j)}(t^* + t) + w_{i,j}e^{-t/\tau} \tag{3.11}$$

For $t = X^*_{(j)}$ we have $\widetilde{V}^{(j)}(t^* + X^*_{(j)}) = \theta$ and from (3.11) we have :

$$V^{(j)}(t^* + X^*_{(j)} + t) = (\theta + w_{i,j}e^{-X^*_{(j)}/\tau})e^{-t/\tau} + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau} I_e^j(s + t^* + X^*_{(j)})\, ds + \frac{1}{\tau}\int_0^t e^{(s-t)/\tau}\sigma\, dW_s \tag{3.12}$$

**Remark.** To find this result we could have integrated the difference between $V$ and $\widetilde{V}$ using the linearity of the model. We keep this simple but longer proof because it is more general and applies to the other results we state in the sequel.

It is clear from equation (3.11) that the hitting time of the barrier $\theta$ by the process $V^{(j)}$, conditionally on the random variable $X^*_{(j)}$ is the sum of $X^*_{(j)}$ and an independent random variable whose law is equal to the hitting time of the barrier $\theta$ of the process (3.8) with initial condition $V^{(j)}(0) = \theta + w_{i,j}e^{-X^*_{(j)}/\tau}$ and with the time shifted input current $\tilde{I^j_e}(t) := I^j_e(t + t^* + X^*_{(j)})$.

$$\eta_{i,j}(u) := \inf\left\{t > 0;\ \ U^{(j)}(t) = \theta | U^{(j)}(0) = \theta + w_{i,j}e^{-X^*_{(j)}/\tau}\right\} \tag{3.13}$$

where $U^{(j)}(t)$ is the solution of equation (3.8) with the time-shifted current specified.

The problem of the first hitting time of the LIF neuron with constant or curved boundaries have been addressed in [24]. We have seen that no closed-form solution can be given to this problem, but many computational methods can be used in order to characterize these hitting times. For instance Volterra's, Durbin's and Girsanov's method for curved boundaries and when the input current is constant the Laplace transform of this random variable is known.

An important remark is that this random variable only depends on $X^*_{(j)}$. Conditionally to $X^*_{(j)}$, the random variable added is independent of the past of the process, so the sequence $X^{(j)}$ is Markovian. Furthermore, the network's countdown process dynamics is autonomous: we do not need to refer to the underlying membrane's potential to describe its evolution. This is very interesting since we can study and simulate this random variable by itself. Therefore, the variable $(X_t)$, possibly added with the time $t$ if the input current is not stationary, is a Markov process, and this process sampled at the times of the spikes is a Markov chain. Furthermore, the law of the zeros of this process is equal to the one of the spikes of the underlying network.

### 3.2.1 LIF model with general post-synaptic current pulse

In this section we consider a LIF neuron described by (3.8). Following the models presented in [14, section 4.1.3], each presynaptic spike generates a postsynaptic current pulse. More precisely, if the neuron $i$ spikes at time $t^*$ and $j \in \mathcal{V}(i)$ receives the spike, then this neuron feel an additional input current given by:

$$I_{PSP}(t^* + t) = w_{i,j}\,\alpha(t) \tag{3.14}$$

Let's include this effect inside the input current $\widetilde{I_e}$ (i.e.

$$\widetilde{I_e}(t) = Ie(t) + \sum_{i \neq j} \sum_{t^j \leq t^j_i \leq t} w_{i,j}\,\alpha(t - t^j_i)$$

where $t^j$ denotes the time of the last spike emitted by the neuron $j$ and $t^j_i$ the sequences of spikes emitted from the neuron $i$ to the neuron $j$.

The same calculations lead to:

$$V^{(j)}(t^* + X^*_{(j)}) = \theta + w_{i,j} e^{-X^*_{(j)}/\tau} \int_0^{X^*_{(j)}} \alpha(s) e^{s/\tau} ds \tag{3.15}$$

So eventually the lasting time to spike for the neuron $j$ is the time the stochastic process $V^{(j)}$, beginning from the value (3.15), reaches the threshold $\theta$, with a new external current. The random variable here is again $(X(t), \widetilde{I}_e(t))_t$. Adding this new Markov process allows us to consider an extended model of the hourglass model in which one component is the countdown process.

In the general case, the countdown process cannot be considered as a Markov process since its dynamics depends on the whole sequence of spikes in the network untill time $t$. Nevertheless this analysis can be simplified if considering postsynaptic current pulses solutions of an ordinary differential equation. This is a very general case, and covers most of the usual models of synaptic coupling (see for instance the works of van Rotterdam and colleagues for the modeling of postsynaptic current pulses [30]). These postsynaptic pulses are in general considered as an exponentially decaying pulse, when taking into account only the decay of the synaptic integration and considering the rise time null. In that case the postsynaptic pulse has the form:

$$\alpha(s) = k e^{-s/\tau_s} \mathbb{1}_{s \geq 0}$$

which is solution of a first order linear differential equation. An even more realistic model taking into account the rise time $\tau_r$ of the synapse and its decay $\tau_s$ is modelled by the following $\alpha$ function for $\tau_r \neq \tau_s$

$$\alpha(s) = \frac{k}{\tau_s - \tau_r} \left[ e^{-s/\tau_s} - e^{-s/\tau_r} \right] \mathbb{1}_{s \geq 0}$$

and for $\tau_r = \tau_s$,

$$\alpha(s) = k\, s\, e^{-s/\tau_s} \mathbb{1}_{s \geq 0}$$

In that case the pulse is solution of a second order linear differential equations.

To take into account this synaptic integration of spikes in our framework, we have to extend the phase space of our Markov chain. More precisely, the markovian model we consider includes a second variable, the spike-induced current $(I^i_a(t))_{i=1\ldots N, t\geq 0}$. If we denote by $\mathcal{L}$ the linear differential operator of the $\alpha$ function, the spike induced current is solution of the equation

$$\mathcal{L} I_a = 0.$$

The new membrane potential equation for a given neuron $i$ in the network is now given by:

$$\begin{cases} \tau_i dV^{(i)}_t & = (-V^{(i)}_t + I_e(t))\, dt + I^{(i)}_a(t)\, dt + \sigma_i dW_t \\ \mathcal{L} I_a(t) & = 0 \end{cases}$$

The Markovian variable we consider is now the process $(X_t, I_a(t))_{t\geq 0}$. When a neuron $i$ elicits a spike, i.e. when its countdown reaches 0 at time $t^*$, its countdown value is reset by drawing in the law of the first hitting time of the membrane potential with initial condition $(V_r, I_a^i(t^*))$ to the threshold and for all neuron $j \in \mathcal{V}(i)$, their spike-induced current $I_a^j(t^*)$ are instantaneously updated by adding the synaptic efficiency $w_{ij}$ : $I_a^j(t^*) = I_a^j(t^{*-}) + w_{ij}$. Simulating this Markov process, that can be sampled at the times of the spike emission, is equivalent from the spikes point of view as simulating the whole membrane potential process.

### 3.3 LIF model with exponentially decaying synaptic integration

In this case we take into account the decay time of synapse at the level of the noise integration itself. In this case, introduced in [24] and whose statistics have been approximated in [25], the membrane potential and the synaptic noise are coupled via the following differential equation:

$$\begin{cases} \tau_i dV^{(i)} &= (\mu_i - V^{(i)}(t))dt + I_e^i(t)dt + I_s^i(t)dt \quad 1 \leq i \leq N \\ \tau_s dI_s^i &= -I_s^i(t)dt + \sigma_i dW_t^i \end{cases} \tag{3.16}$$

and the spiking condition reads:

$$V^{(i)}(t^-) = \theta \Rightarrow \left\{ \begin{array}{lcl} V^{(i)}(t) &=& V_r \\ I_s^j(t) &=& I_s^j(t^-) + w_{i,j}\mathbb{1}_{j\in\mathcal{V}(i)} \end{array} \right. \tag{3.17}$$

Qualitatively, when a spike is received by a neuron, the synaptic current $I_s$ integrates the spike and the effect on the membrane potential is smoother. Therefore in this model it is interesting to consider post-synptic pulses having the same dynamics as the noise integration, i.e. solution of the differential equation:

$$\tau_s \frac{\mathrm{d}I_a}{\mathrm{d}t} = -I_a(t).$$

The very same analysis could be done if we considered an instantaneous spike integration, but we do not present the results here since it seems strange to us to consider to levels of synaptic integration: the noise integration and the spike integration. The calculations can nevertheless be driven exactly as in the case of the perfect integrate-and-fire.

The reset random variable is given by the first hitting time of the membrane's potential stochastic process. Driving the same type of calculus as in the previous section we obtain, for $j \in \mathcal{V}(i)$ and $\tau \neq \tau_s$ the relationship :

$$V^{(j)}(t^* + t) = \widetilde{V}^{(j)}(t^* + t) + e^{-t/\tau} w_{i,j} \frac{1 - e^{-\alpha t}}{\alpha} \tag{3.18}$$

where $\alpha = \frac{1}{\tau_s} - \frac{1}{\tau}$ and again $\widetilde{V}^{(j)}(t^* + t)$ the membrane potential of the neuron $j$ without any interaction. We can see that after the time $X^*_{(j)}$, the membrane potential of $j$ is $\theta + w_{i,j}e^{-t/\tau}\frac{1-e^{-\alpha X^*_{(j)}}}{\alpha}$. The evolution of the potential $V^{(j)}$ after $t^* + X^*_{(j)}$ and conditionally

on $X^*_{(j)}$ and $I_s(t^*)$ is independent of the past, so we have to wait for the process (3.17) to reach the threshold $\theta$ from the initial condition $\theta + w_{i,j}e^{-t/\tau}\frac{1-e^{-\alpha X^*_{(j)}}}{\alpha}$ and with the "time and space" shifted currents $\tilde{I^j_e}(t) := I^j_e(t+t^*+X^*_{(j)}) + w_{ij}e^{-\frac{X^*_{(j)}}{\tau_s}}$. In the case $\tau = \tau_s$ we only have to replace the expression $\frac{1-e^{-\alpha X^*_{(j)}}}{\alpha}$ by $X^*_{(j)}w_{i,j}e^{-t/\tau}$, and the change in the currents is the same.

Therefore, the variable $(X_t, I_s(t))$ is Markovian and we deduce the precise firing times from its study. This Markovian variable necessitates to evaluate the law of the first hitting time of a DIP to a curved boundary, which can be achieved using the technique provided in [25]. As in the case of the perfect integrate-and-fire neuron, we can show that this process, eventually augmented with the time $t$, and possibly sampled at the times of the spikes, satisfy the Markov property and the law of the zeros of the Markov process is the same as the law of the spikes of the underlying network. Indeed, we have seen that the coutdown process was no more a Markov chain. In order to compute the reset random variable, we need to consider the value of the synaptic current at the spike time. Furthermore, for non stationary inputs, we have to add the time as a new variable. The dynamics of this process can be described as follows: consider that this process after the $n^{\text{th}}$ spike is $(X^n, I^n_s, t^n)$. Then the next spike will be fired from the neuron $i_n$ having the lowest countdown value. It will fire at time $t^{n+1} = t^n + X^n_{i_n}$. Its countdown value will be reset to the first hitting time of the related DIP, and proved in [25], the law of the pair composed of next spike time and the relative location of the synaptic currents at this time is known. Therefore by drawing in the law of this pair, we have thus the new countdown value and the future synaptic current at the time of the next spike for $i_n$. Similarly, each neuron $j \in \mathcal{V}(i)$ is updated according to the law of the first hitting time of the related membrane's potential starting with input current given by $I^n_j$ to reach a given threshold, and therefore the new countdownw value and the future location of the input current are computed at the same time using the results of chapter [25]. The other neurons' state are unchanged. It is clear that the law of the spikes is the same as the law of the zeros of the countdown process.

## 3.4 LIF models with noisy conductances

The interactions considered in the last subsection are reasonable models of current interaction. Nevertheless reality it is even more complicated. Indeed, the effect of a spike on the postsynaptic cell does not directly results in the generation of currents. It results in changes in the membrane's conductance, and these modifications produce a ionic current. This resulting current is approximately proportional to the membrane's voltage potential. The modulation of the conductance of the post-synaptic membrane has a certain time course $g(t-t^*)$, which is in general considered as constant, to keep the model tractable. Here again we consider the noise and the spikes integrated in the same fashion, i.e. via the conduc-

tances. Therefore the membrane potential when no spike is received is solution of the linear stochastic differential equation:

$$\left\{ \begin{array}{lcl} dV^{(i)} & = & (I_e(t) - \lambda(V_t^{(i)} - V_{rev}))\, dt + I_s(t)\, dt + \sigma_i\, g_i\, (V^{(i)} - V_{rev})\, dW_t^i \\ V^{(i)}(t^-) & = & \theta \Rightarrow V^{(i)}(t) = V_r \end{array} \right. \tag{3.19}$$

In this equation the term $I_s$ corresponds to the current generated by the spikes. When neuron $j$ receives a spike from one of its neighbors $i$, a current $I_s$ is generated, which has the value $w_{ij}g(V^{(j)} - V_{rev})$ ($V_{rev}$ is the reversal potential of the synapse). Note that we artificially introduced $V_{rev}$ in the leak term, which amounts to formally changing the current $I_e$, in order to integrate more simply this equation. We clearly see in this equation the effect of a presynaptic spike on the conductances. More precise models take into account the vanishing of this effect in the time. General time profiles of the postsynaptic conductance pulses are alpha functions as described in the previous section, and hence can be modeled as solution of a linear ordinary differential equation of order one, two or greter. Nevertheless, even in the simpler case, we will see that these models cannot be expressed as a Markovian model in function of the countdown process and possibly other real processes.

We first consider the case where the neuron $j$ receives a spike at time $t^*$ from neuron $i$ and that this increases the conductance by a coefficient $w_{ij}\, g$. The solution of the membrane potential's equation after time $t^*$ reads:

$$V^{(j)}(t + t^*) = V^*_{(j)} Z_t + \int_0^t I_e(s + t^*) Z_{t-s}\, ds \tag{3.20}$$

where $Z_t = \exp\{-(\lambda + 1/2\sigma^2 - w_{ij}g)(t - t^*) + \sigma W_t\}$. The membrane potential if no spike were received at time $t^*$ would read:

$$\widetilde{V}^{(j)}(t + t^*) = V^*_{(j)} \widetilde{Z_t} + \int_0^t I_e(s + t^*) \widetilde{Z_{t-s}}\, ds \tag{3.21}$$

where $\widetilde{Z_t} = \exp\{-(\lambda + 1/2\sigma^2 - w_{ij}g)(t - t^*) + \sigma W_t\} = e^{w_{ij}g(t-t^*)} Z_t$. At time $X^*_{(j)}$ the membrane potential reads:

$$V^{(j)}(X^*_{(j)} + t^*) = \theta e^{w_{ij}\, g\, X^*_{(j)}} + \int_0^{X^*_{(j)}} I_e(s + t^*) Z_{t-s} (e^{w_{ij}\, g\, s} - 1)\, ds$$

This expression therefore depends on the whole past of the Brownian motion, and cannot be written as a function of $X^*_{(j)}$, even taking into account the conductance as an additional variable. These models will not be subject to the markovian modelization we propose in this paper.

# 4 Balanced networks with synaptic delays and refractory period

In the previous section, the instantaneity of the interaction at the level of the synapse lead us to consider only inhibitory interactions in order to avoid the contradictory problem of spike avalanche we described above. In biological network, this issue does not appears because the refractory period of the neuron upperbounds the firing frequency. Furthermore, the synaptic delays in the transmission of the spike also avoids the avalanche process by avoiding the self-excitation, through the network, of a given cell. The refractory period is a transient phase just after the firing during which it is impossible very difficult to excite the cell. This phenomenon is linked with the dynamics of ion channels and the hyperpolarization phase of the spike emission, lasts few milliseconds, and prevents the neuron from firing spikes at an arbitrary high firing rates. It can be decomposed into two phases: the *absolute* refractory period, which is a constant period of time corresponding loosely to the hyperpolization of the neuron during which is it impossible to excite the cell no matter how great the stimulating current applied is (see for instance [16, chapter 9] for a further biological discussion of the phenomenon and [14, 1] for a discussion on modelling this refractory period). Immediately after this phase begins the *relative* refractory period during which the initiation of a second action potential is inhibited but still possible. It amounts considering that the synaptic inputs received at the level of the cell are weighted by a function depending on the time elapsed since the spike emission. This phase also lasts around one millisecond.

To be coherent in our modelization, when we take into account such fast phenomena, we need to consider in another addition fast phenomenon: the axonal spike transmission from the presynaptic cell to the postsynaptic one. The delay induced by the spike transport and its transmission via the synapse depends on the distance between the two cells, the speed of transmission of the signal along the axon and the transmission time at the synapse, and has a typical duration of few milliseconds.

To model the absolute refractory period, we consider that if the neuron indexed by $i$ fires at time $t$, it stays at his resting potential $V_r^{(i)}$ untill time $t + R_i$ where $R_i$ is the time duration of the absolute refractory period, that only depends on the presynaptic neuron.

We model the relative refractory period only for the spike integration, and not for the noise integration. Indeed, the stochasticity of the membrane potential does not exclusively comes from the synapses, and therefore might not be influenced by the state of the ion channels. Moreover, the noise coming from the uncorrelated activity at the synapse creates also a Brownian current, which is very small in law (the probability of the integrated process at the level of the membrane during a time period as short as 1 or $2ms$ to be substantial is very small). Therefore we consider that the integration of the noise is not influenced by the relative refractory period. For the network's interaction, this remark is no more valid: the informations do transit via the synapses, and the change of membrane potential is consistent. These synaptic efficiency will be weighted by a function depending on the time elapsed since the last spike has been fired. We denote this function $\kappa(t)$ following the notation of Gerstner and Kistler in [14]. In our case this function is unspecified, is zero at $t = 0$ and increases to

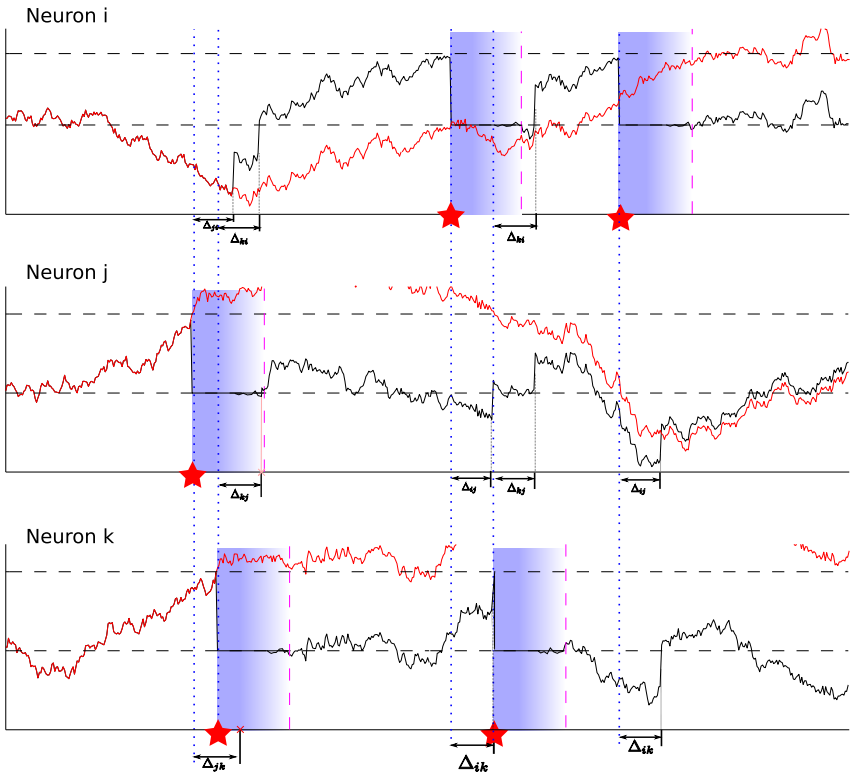



Figure 4: Sample path of the membrane potential of a three-neuron excitatory network with synaptic delays and refractory period in the case of the perfect integrate-and-fire neuron. The black curve represents the membrane potential, the red curve the membrane potential's process with no reset and no interaction. The spikes are represented by red stars and blue dotted lines. The refractory period is represented by the blue boxes: plain blue for the absolute refractory period, and the intensity is proportional to the attenuation of the spike during the relative refractory period.

1 with a characteristic time of around $2ms$. It can be of bounded support of defined other $\mathbb{R}$, but it will very fast be very close to 1.

To model the synaptic delay we consider that spikes emitted by a neuron do not affect instantaneously the target neurons, but only after some delay $\Delta_{i,j}$ which can depend on both the presynaptic and the postsynaptic neurons (see figure 4) since this delay is linked to the duration of the spike transmission and therefore may depend on some measure of distance between the pre- and post-synaptic neurons.

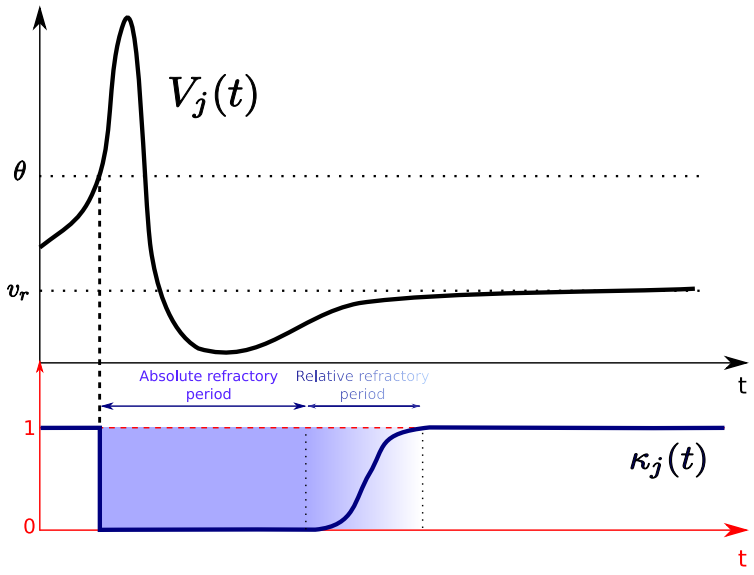



Figure 5: The refractory period at a spike emission, and the related $\kappa$ function weighting the synaptic inputs

In the present section, we model these three phenomena, and show that in these cases we can also define a Markov chain describing the spike times of the neural network, that will be also based on the hourglass model. For inhibitory networks, the modifications with the previous framework is very simple, but it will become slightly more complex, but still tractable, when taking into account synaptic delays. For the sake of compactness of notations, we define the function $\kappa_j(t)$ for all $t > 0$. This function is identically equall to 0 for $t \in [0, R_j]$, and increases to one after time $R_j$ with a characteristic time of the order of the millisecond (see figure 5. If neuron $i$ fires a spikes at time $t_i$, its effect on the postsynaptic neuron $j$ depends on the synaptic delay $\Delta_{i,j}$, the countdown value $X^{(j)}(t)$, and the time of the last spike emitted by $j$:

(i). If $\Delta_{ij} < X^{(j)}(t)$, then the reception of a spike at time $t$ acts on the post-synaptic neuron at time $t + \Delta_{i,j}$ in the same fashions as discussed in the different models considered in section 3, but in that case the interaction can be either excitatory or inhibitory, with a synaptic efficiency $w_{ij}\kappa_j(t_i + \Delta_{ij} - t_j)$ .

(ii). If $\Delta_{ij} > X^{(j)}(t)$, the postsynaptic neuron will fire before receiving the spike from the presynaptic cell $i$, and it will act on the postsynaptic cell's membrane with an efficiency $w_{ij}\kappa_j(t_i + \Delta_{ij} - X_j)$.

Let us consider the effect of these features from the viewpoint of the countdown process. The reset variable is only affected by the absolute refractory period, and in a very simple way. Indeed, we formally consider that the neuron $i$ is stucked at its reset value $V_r^{(i)}$ during a period of time $R_i$ after having fired. After this period, the neuron's membrane potential follows its evolution depending on the model chosen. Therefore, time of the next spike starting from time $t + R_i$ has the law of the reset variable in the case where we did not take into account refractory period and synaptic delay, i.e. has the law of the first hitting time of the membrane potential process to the spike threshold, with the time-shifted input $I_e(t + t^* + R_i)$ in the case of non-stationary inputs. If we denote $\tau_i$ this random variable, the new reset variable of the related countdown process has simply the law of $Y_i = \tau_i + R_i$.

The case of the interaction variable is a little bit more intricate, and we will deal with it in the following subsections.

## 4.1 Modeling the refractory period

We first consider that the transmission delay is null. In this case the effect of a presynaptic spike on the cell $j$ will be weighted by the function $\kappa_j(t-t_j)$ where $t_j$ is the time of the last spike emitted by the cell $j$. We show that for the models discussed in section 3, the spikes in the network have the same law as the zeros of a simpler Markov process based on the countdown process, and that this dynamics can be reduced to the one of a Markov chain. To this purpose, we identify the random variables needed to define the countdown process. In figure 6 we repesented a sample path of the countdown process related to the perfect integrate-and-fire neuron when considering an absolute refractory period. Two random variables are necessary to define the dynamics of the countdown process: the reset variable and the interaction variable. We already identified the law of the reset random variable for the countdown process when considering a refractory period. For the interaction variable, the case is readily deduced from the analysis of section 3 in the case of inhibitory interactions. Indeed, if neuron $i$ elicits a spike at time $t_i$, it will affect the postsynaptic neuron $j$ only if it is not during its absolute refractory period, with a weighted synaptic efficiency. Denote by $t_j$ the time of the last spike elicited by $j$. The presynptic spike coming from neuron $i$ will affect the neuron $j$ only if $t_j + R_j < t_i$, and if it does, the action of the presynaptic spike on the next spike time has exactly the same effect as treated in the previous section, with a synaptic efficiency $w_{ij}\kappa_j(t_i - t_j)$. Therefore, adding a refractory period makes the random variable depend upon the last firing times of each neuron.

To take into account this fact, we define the *last firing times* variable $H \in \mathbb{R}^N$, that stores the last spike time of each neuron. All its components are set to $R \stackrel{\text{def}}{=} \min_{i=1\dots N} -R_i$ at the initial time. The $j^{\text{th}}$ component is this variable is constant between two spikes of the neuron $j$. If neuron $j$ spikes at time $t_j$, this component is instantaneously set to $t$, and all the other components of this variables are unchanged. This value will remain constant untill neuron $j$ spikes again[3].

[3]this variable could also be modelled as a renewal process that is set at each spike time of neuron $i$ at the value $R_i$, and the neuron $i$ can receive stimulations from its presynaptic spikes only when this value is 0.

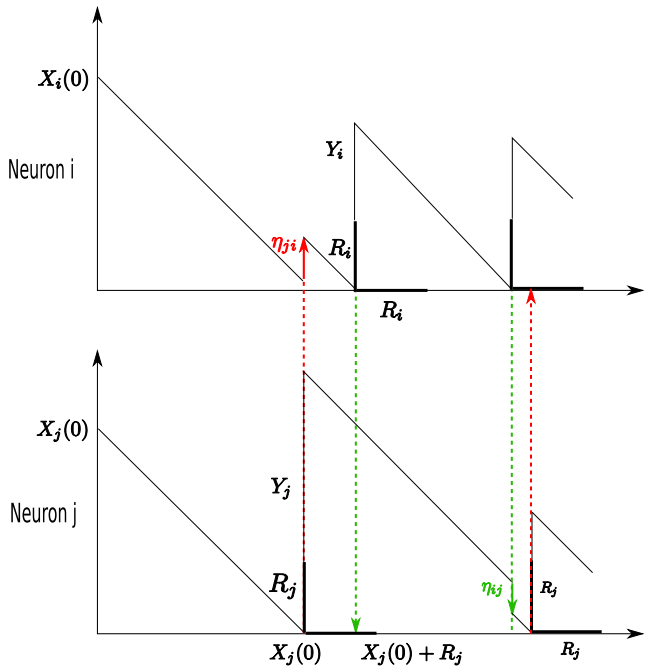



Figure 6: A sample path of the countdown process taking into account the refractory period. This figure represent the countdown value for a 2 neurons network. Neuron $j$ fires first at an instant where the cell membrane is excitable, and has an inhibitory effect on neuron $i$, whose spike is posponed. When neuron $i$ fires for the first time, the neuron $j$ is still in its refractory period and therefore does not integrate the effect of the incoming spike. The second spike emitted from neuron $i$ excites the cell $j$ and advances the spike time. Neuron $j$ then spikes during the refractory period of neuron $i$.

If the synapses are inhibitory, the interaction variable $\eta_{ij}$ of the new countdown process is simply deduced from the law interaction variable $\tilde{\eta}_{ij}(w_{ij})$ by changing the synaptic weight. In the case of realistic refractory period, we have $\eta_{ij} = \tilde{\eta}_{ij}(w_{ij}\kappa_j(t_i - t_j))$. In the particular case of pure absolute refractory period, this random variable is simply $\eta_{ij} = \tilde{\eta}_{ij}(w_i j)\mathbb{1}_{X_i > H_j + R_j}$. This new interaction variable depends therefore on the same variables as the one in the case of section 3 and on the last firing time variable. The process $(X_t, H_t, t, A_t)$, where $A_t$ are the possible additional variables (typically the value of the synaptic current or the interaction current) is hence a Markov process. Indeed, if all the components of $X$ are strictly positive, the time increases linearly with slope 1, the countdown process decreases linearly with slope $-1$, the last firing time variable and the possible additional variables remain constant. If the component $i$ of the countdown process reaches 0, this neuron spikes. Almost surely only one neuron realize this infimum at a given time. At this time, say $t_i$, the following operations occur:

- $X^{(i)}(t_i)$ is reset to an independent copy of $Y_i$ and $A_{t_i}$ is updated according to its dynamics.
- $H_i(t_i) = t_i$ and $H_j(t_i) = H_j(t_i^-)$ for all $j \neq i$ (i.e. theses components are unchanged).
- $\forall j \in \mathcal{V}(i)$, $X^{(j)}(t_i) = X^{(j)}_{t_i^-} + \eta_{i,j}\mathbb{1}_{t_i > H_j + R_j}$
- the time is trivially updated.

After this phase, the dynamics proceeds the same way.

For excitatory interactions, the case is slightly more complex. The previous calculations are valid only in the inhibitory case, since we used the Markov property of the processes we were studying to compute the interaction variable. More precisely, when an inhibitory interaction occurs, the time of the next spike is increased. The state of the countdown value gave us the time of the next spike, together with the state of possible additional variables. This information was taken into account: using the Markov property of the processes we studied, we stated at the time of the expected spike if no interaction had taken place meanwhile, and from this point we computed the law of the additional time to wait untill the next spike because of the interaction. In the case of excitatory interactions, this trick cannot be applied: indeed, the time to the next spike after the excitatory interaction is smaller than the one predicted by the countdown process. When we were conditionning on the past in the case of inhibitory interactions, we will be conditionning on the future in the case of excitatory synapses in order to derive our random variables. This is not a big deal conceptually, but we have to be careful when deriving these random variables.

**Perfect IF neuron with instantaneous synapses:** Assume that the neuron $j$ receives a spike from neuron $i$ at time $t_i$. The countdown process value of the neuron $j$ just before this interaction is denoted $X_j^*$. The interaction random variable $\eta_{ij}$ is the difference of time between the spike time after interaction and the spike time before interaction, conditionnaly to the fact that this next spike time was predicted to be $X_j^*$. After some simple calculations,

we observe that it has the law of the first hitting time of the membrane potential process to $\theta - w_{ij}$ conditionnaly to the fact that the first hitting time of this process to $\theta$ is $X_j^*$. Denote by $\zeta_{ij}$ this random variable. The law of the update random variable $\eta_{ij}$ will be defined by $(\zeta_{ij} - X_j)\mathbb{1}_{X_i > H_j + R_j}$ (note that the variable $\zeta_{ij}$ is always positive; if the interaction makes the neuron spike instantaneously, its means that $\zeta_{ij} = 0$ and therefore the new countdown value for $j$ is 0). Furthermore in that case, since the Markov's property cannot be used, the random variable will not be independent of the value of the membrane's potential at the time of the spike, which we denote by $V_j^*$. Let us characterize the law of $\zeta_{ij}$:

$$\begin{aligned}
\mathbb{P}\left[\zeta_{ij} = u\right] &= \mathbb{P}\left[\tau_{\theta - w_{ij}\kappa_j(t_i - H_j)} = u \middle| V_j^*, \tau_\theta = X_j^*\right] \\
&= \mathbb{P}\left[\tau_\theta = X_j^* \middle| V_j^*, \tau_{\theta - w_{ij}\kappa_j(t_i - H_j)} = u\right] \frac{\mathbb{P}\left[\tau_{\theta - w_{ij}\kappa_j(t_i - H_j)} = u \middle| V_j^*\right]}{\mathbb{P}\left[\tau_\theta = X_j^* \middle| V_j^*\right]} \\
&= \mathbb{P}\left[\tau_\theta = X_j^* \middle| W_u = \theta - w_{ij}\kappa_j(t_i - H_j)\right] \frac{\mathbb{P}\left[\tau_{\theta - w_{ij}\kappa_j(t_i - H_j)} = u \middle| V_j^*\right]}{\mathbb{P}\left[\tau_\theta = X_j^* \middle| V_j^*\right]} \qquad (4.1)
\end{aligned}$$

This random variable is null whenever $V_j^* > \theta - w_{ij}$. This gives us the law of the interaction variable in the case of excitatory inputs. Nevertheless, we can see that it involves the value of the membrane potential's process at the times of the spike. Therefore, we need an additionnal variable in order to define autonomously the countdown process: it is the membrane potential's value at the times of the spike receptions. At each time that a spike is emitted in the network, this variable is updated in the following fashion:

- For the neuron that elicited a spike, this value is set to $V_r$
- For the other neurons, it is updated by drawing in the law of the membrane potential conditionally to the fact that it will reach the threshold at the time given by the countdown process.

In summary, to simulate the process with a Markovian framework including the countdown process, we simulate a discrete time Markov chain $(X, H, V)$, where $H$ is the last firing time variable, $X$ the countdown process and $V$ the membrane potential at the time of the spike. The transition of this chain from $(X^n, H^n, V^n, t^n)$ is given by:

$$\begin{cases}
V_{i_n}^{n+1} & = V_r \\
V_j^{n+1} & : \text{ drawn in the law of } V_t^{(j)} \text{ conditionnaly to the fact that it is} \\
 & \quad V_j^n \text{at time } t^n \text{ and } \theta \text{ at time } t^n + X_j^{(n)} \text{ for } j \neq i_n \\
t^{n+1} & = t_n + X_{i_n} \\
H_{i_n}^{n+1} & = t^{n+1} \\
X_{i_n}^{n+1} & = Y_{i_n} \\
X_j^{n+1} & = X_j^n + \eta_{i_n j}(X_j^n, V_j^{n+1}, H_j^{n+1}) \text{ for } j \neq i_n
\end{cases}$$

**Excitation for perfect IF neuron with synaptic integration:** In the case of excitatory synapses, the same issue as before appears: the Markov property does not apply, and hence we have to apply the same transformation as we performed in equation (4.1). In that case, the calculations lead to keep in memory both the membrane potential and the synaptic current at the times of the spikes. The same type of expressions and the same type of dynamics of the resulting Markov chain is obtained. Indeed, assume that the countdown process value at the time of the spike is $X_j^*$ and the value of the additional variable (the synaptic current at the time of the next spike) is $I_j^*$. Then the probability to spike at time $u < X_j^*$ and for an input current $I_s = v$ after the excitation has been received can be computed as the first hitting time of the underlying membrane potential. In the case of instantaneous interactions, it has the law of the first hitting time of the membrane potential process starting at $(V_j(t^*), I_j(t^*))$ the values of the membrane potential and of the synaptic current at the time of the spike considered, to reach the threshold $\theta - w_{ij}\kappa_j(t_i - H_j)$ at time $u$ with the synaptic current $v$ conditionnaly on reaching the threshold $\theta$ at time $X_j^*$ with the input current $I_j^*$. The law of this random variable can be computed in the same fashion as we did in (4.1). We can see that it depends on the value of the membrane potential and of the input current at the times of the spikes. The law of this random variable is known and can be computed. Therefore, we can provide a Markovian framework to study this type of behaviors. For the other types of perfect integrate-and-fire models, the same reasonning applies and we get blou blou....

**LIF with instantaneous synapses** In the case of the leaky integrate-and-fire neuron with instantaneous synaptic integration, no further simplification can be provided, and we obtain that the new spike time after interaction has the law of the first hitting time of the membrane potential process to reach the boundary $\theta - w_{ij}\kappa_j(t_i - H_j)e^{-t/\tau}$ (where $t_i$ is the time of the presynaptic spike) conditionnaly on the fact that the first hitting time of the boundary $\theta$ is equal to $X_j^*$. In that case again, we need an additional variable: the membrane potential at the times of the spikes, in order to define a Markov chain containing the times of the spikes.

The case of postsynaptic current pulses can be treated in the same way. In that case again it will be necessary to know the membrane potential's voltage at the times of the spike in order to be able to simulate the countdown process.

**LIF with synaptic integration** The case of the LIF neuron with synaptic integration can be treated in the same fashion as the case of the perfect integrate-and-fire with synaptic integration.

## 4.2 Including synaptic delays

When we include the synaptic delays in addition to the refractory period, the reset variable of the related countdown process is the same as in the case where we only consider the cell's refractory period: taking into account the axonal delay does not affect the reset variable

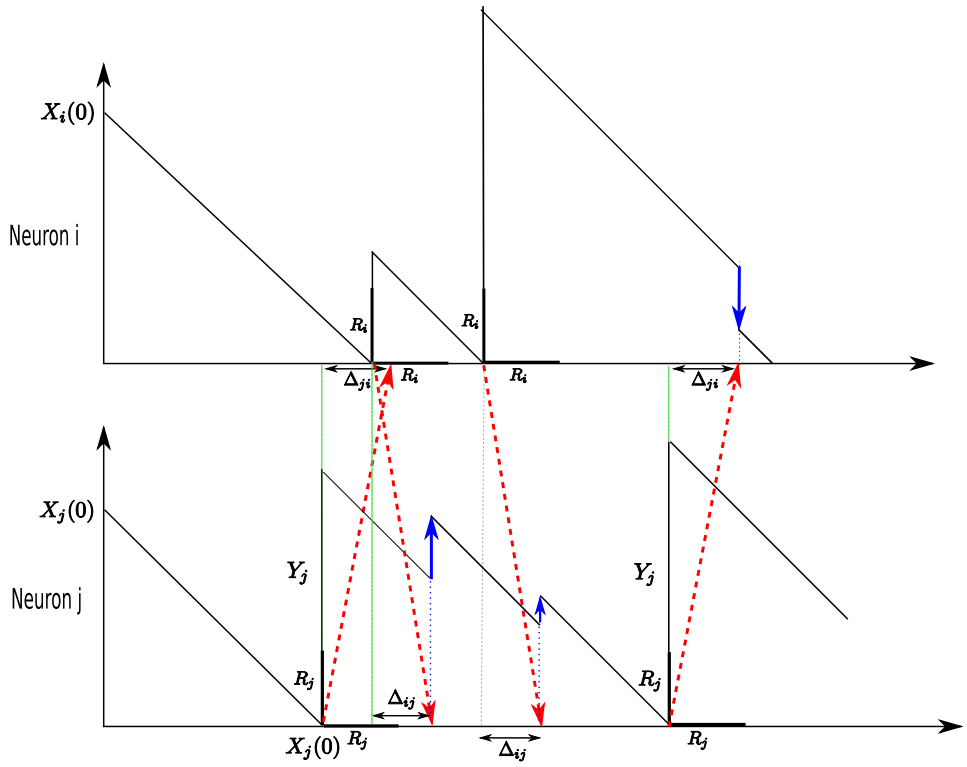



Figure 7: A sample path of the countdown process in the case of the perfect integrate-and-fire neuron with instantaneous interactions, when taking into account the synaptic delays and the refractory period. The first spike is emitted by neuron $j$ but arrives at neuron $i$ during its refractory period hence does not affect its evolution. Neuron $i$ sends a spike during the refractory period of neuron $j$ which is received after this period and hence affects the dynamics of the countdown process. The action of neuron $i$ on $j$ is inhibitory and the action of $j$ on $i$ excitatory.

which, for a given neuron $i$, has the same law as $R_i + \tau_i$ where $\tau_i$ has the law of the first hitting time of the membrane potential process to the threshold starting from $V_r$. The synaptic delay nevertheless affects the interaction variable in a quite intricate fashion. A sample path of the countdown process in that case is represented in figure 7. Nevertheless, it adds a non-trivial memory-like phenomenon in the network. Indeed, since spikes do not reach instantaneously the postsynaptic neuron, the postsynaptic neuron can fire, be reset, and then integrate a previous spike. This changes our framework: indeed, in our framework, the random variables were updated instantaneousluy at each spike time, even when the

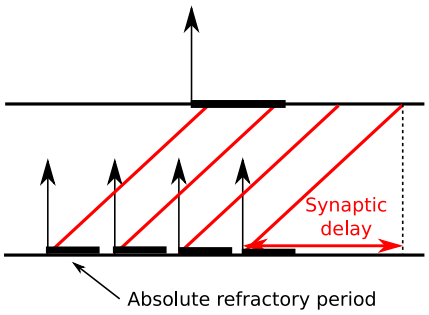



Figure 8: Presynaptic spikes emitted before a postsynaptic spike can affect the postsynaptic cell after the firing.

interaction at the level of the membrane potential was not. The delay creates a network memory that leads us to keep in memory a certain number of spikes. Fortunately, because of the absolute refractory period, we only have to take into account a finite number of spikes that can possibly affect the postsynaptic potential after it elicits a spike (see figure 8). The maximal number of spikes concerned is given by $M \stackrel{\text{def}}{=} \lfloor \frac{\Delta_{ij}}{R_j} \rfloor$ where $\lfloor x \rfloor$ is the floor function, i.e. the largest integer small or equal to $x$.

In this case, instead of considering the last firing times variable which contained only the last firing time for each neuron of the network, we consider the last $M$ firing times variables. This variable is a matrix $H_M \in \mathbb{R}^{N \times M}$. Each line of the matrix corresponds to the $M$ last firing times of the neuron. Each line $i$ of the matrix is constant between two spike times of the neuron $i$. At the initial time, the $M$ components of this line are set to the value $\min_{i,j}\{-R_i - \Delta_{ij}\}$. If the neuron spikes at time $t_i$, then each component of the line are modified: for all $k \in \{2, \dots, N\}$, $H_{i,k-1} = H_{i,k}$ and $H_{i,M} = t_i$. This matrix stores the times of the $M$ last spike times of the neuron $i$, in the chronological order.

In this case again, we can describe a Markov process and a Markov chain in order to reproduce the times of the spikes. This chain is composed of the same elements as the model with no delay. Let us denote by $X^n$ the countdown chain , by $A^n$ the possible additional variables, $H^n$ the $M$ last firing times variables, $t^n$ the event times, and by $\tilde{V}^n$ the variables containing the membrane potential at the times of the spikes, which is necessary only in the case of excitatory interactions. An event in this chain is either a spike, or the arrival of a spike on a postsynaptic spike, now that these two events are no more simultaneous. The next spike if no delayed interaction occurs will be fired after a time given by $\tau = \min_i X_i^n$, and the first arrival of a possible spike at a cell is given by

$$\nu = \min_{\substack{i,j \in \{1, \dots, N\} \\ k \in \{1, \dots, M\}}} \{x = H_{i,k} + \Delta_{ij} - t; x > 0\}$$

If this set is empty, the min is set to $+\infty$. If $\tau < \nu$, a spike will be fired by the neuron $i$ having the lowest countdown value. The state of the countdown variable for this neuron is reset according to the law we already described, and all other variables are updated: the line $i$ of the last $M$ firing times will be updated, the time will be updated to $t + X_i$, and the additional variables are updated. No interaction is taken into account at this time. If $\nu < \tau$, assume that the minimum is achieved for the value $H_{i,k} + \Delta_{ij}$ for some $i, j, k$. This means that the $k^{th}$ latest spike of neuron $i$ reaches the cell $j$. Therefore, the related interaction variable of this connection will be added, and the countdown value of neuron $j$ will be updated, together with the possible additional variables. The time is advanced to $t + \nu$. Note also that many 3-uplets $(i, j, k)$ can achieve this min at the same time. Moreover, it is possible also that an excitatory interaction makes a postsynaptic neuron fire instantaneously at the reception of the spike. All these cases might be treated sequentially, by iterating the mechanism we just described. Nevertheless, we are ensured that no avalanche can occur, because of the absolute refractory period and of the delays.

We finally note that in the case of purely inhibitory networks, the update of this chain can be done only at the times of the spike. Indeed, let us consider that the state of this chain at the iteration $n$ is $(X^n, H^n, t^n, A^n)$ and that neuron $i$ just spiked. We then compute its next spike time if no interaction occurs meanwhile $Y_i$. But we know through the variable $H^n$ that possibly, before this time $Y_i$, spikes emitted from other neurons will arrive at the synapses of $i$. We can therefore at this same time $t^n$ draw in the laws of the interaction variables the additional time that their arrival will provoke on the next spike time for $i$ (this time may depend on the time when the spike will arrive at the synapse of $i$, which can also be computed with the variables we have). Therefore in that case, the countdown value will be an hourglass chain as defined in section 1.

# 5 Ergodicity of the network

The approach we developped in the last sections resulted in providing a simpler framework than the usual one based on the membrane potential for modeling the spikes in a neural network of stochastic integrate-and-fire neurons This model is equivalent in law from the viewpoint of the spike times, to the usual model. If this modelization gives us a very natural and sometimes very efficient way for simulating the network (see section 6), it also provides us a good framework for studying its mathematical properties in a more elegant and tractable fashion. Indeed, the models we obtained fit into a class of models studied in the queuing theory in the past ten years. The first analysis of this type of modelled is due to Marie Cottrell [6]. In this article she studied the hourglass model where the interaction random variable is deterministic and inhibitory (i.e. $\forall i, j$, we have $\eta_{i,j}(u) \equiv \eta$ where $\eta$ is a positive constant). In her article, she proves in that case that the related Markov chain is irreductibility and aperiodicity. Furthermore, she provides a criterion for the positive recurrence of this chain and characterizes the ISI for a two neuron network. In the transient case she shows that some neuron will stop firing in a finite time, and study the pattern formed by the "dead" neurons (those that will never spike again).

The proof of the irreductibility and aperiodicity of the chain consists in constructing a set of probability in which all the $N$ neurons fire consecutively. The probability of this set is strictly postive, and we can show that every state is accessible after the $N$th spike triggered by the last neuron. The same analysis can be done after the next spike, so at spike $N+1$, which proves that the embedded Markov chain $X_n$ is irreductible and aperiodic.

She then proves that if $\mathbb{E}\left[Y_i^2\right] < \infty$ and $\eta < \inf_{i=1,\dots,N} \frac{\mathbb{E}[Y_i]}{|\mathcal{V}(i)|}$ where $\eta$ is the interaction constant, $Y$ the reset random variable and $|\mathcal{V}(i)|$ the number of neighbors (postsynaptic neurons) of the neuron $i$, then the countdown process $(X_t)_t$ and the related Markov chain $(X_n)$ are ergodics, irreductibles, aperiodics and positive recurrents.

This result was then generalized by Fricker, Robert et al [13]. In this paper, the authors find necessary and sufficient conditions of ergodicity for the system when the variables of interations $\eta_{i,j}$ does not depend on the state of the variable, and are an iid sequence of random variables (but no more considered deterministic). Assume that the network is fully connected, and that the reinitialisation random variables $Y_i$ are exponentially distributed, with parameter $\lambda_i$, and that the interactions are the same for all the neighbors of a neuron (i.e. $\eta_{i,j} = \eta_i$ for all $j \in \mathcal{V}(i)$).

For the fully connected network, the authors prove that the network is stable if $\rho = max_i \rho_i := max_i \mathbb{E}[\eta_i]\,\mathbb{E}[Y_i] < 1$. Under this stability condition, they give an explicit expression for the Laplace transform of the invariant measure of the Markov process associated to this model. Then they prove that if $\rho > 1$, then the network is not stable, and after a finite time, only one neuron would spike and all the other neuron are "dead" (i.e. will not fire anymore).

Then the authors examine also the case of the linear networks. The interaction variable considered are independant and identically distributed random variables $\eta_i$, which is exponentially distributed with the same parameter $\lambda$. In this framework they prove that:

(i). if $N$ is odd then the network is stable if $\rho = \frac{\lambda}{\mu} < 1/2$ and not stable if $\rho > 1/2$.

(ii). if $N$ is even, then the network is stable if $\rho < \frac{1}{2cos(\pi/(N+1))}$ and not stable if $\rho > \frac{1}{2cos(\pi/(N+1))}$

Note that the proof of ergodicity is based on an adapted version of the *second vector field* associated to a Markov process. It was introduced by Malyshev and Menshikov in [19].

These results were later generalized by Turova. She studied also the effect of excitatory connections. She proved for instance that in a simple balanced networks with iid interactions, there exist a critical value of the ratio excitation/inhibition below which the network is transient and above which the network is recurrent [7]. She also proved for a purely excitatory network that there almost surely existed a time for which all neuron spike (complete synchronization of the network, see [29]) and in another context that adding one inhibitory connection augmented the probability of synchronization, i.e. stabilizes the oscillations of the total activity [26]. She studied then the effect of plasticity in these networks [28], and the type of patterns observed in the case of transient networks, which she relates with neuronal coding [7]. She also opened the way to more realistic models of neurons.

We are interested in the present section in generalizing these results to the case of stochastic integrate-and-fire networks. We will not go into the details, but the model we propose here fits in the framework of these works, and therefore could be studied with the same mathematical tools as the ones used in these previous publications. An interesting way for studying these networks would be to used hydrodynamics limits presented in [9, 10, 8, 17]. Antoher very interesting to study mathematically these models would be the dynamical system approach to networks as developped by Malyshev and collaborators in [19, 12, 18]. These works are outside the scope of this dissertation and still an active research area. Furthermore, the usual questions solved in this framework, such as the ergodicity or the transience, are not of great interest from a biological point of view. Indeed, the questions that naturally arise in neuroscience when studying this type of networks are mostly discriminating between chaos and oscillations and characterizing the temportal features of activity. If we can prove the ergodicity of the network, it will be therefore interesting studying the stationary measures for instance. In this section we focus of the application of the results already proved to the biological cases, not taking into account the excitation.

Indeed, in the case of purely excitatory networks, the chain will always be ergodic provided that the reset random variable is almost surely finite. This property depends on the neuron model we consider and the input current. If this random variable is not almost surely finite, the probability to stop firing for a given neuron in the network is simply equall to the probability that its reset variable is infinite, and no network effect has to be taken into account. The problem of balanced networks having both excitatory and inhibitory connections is more complex. To obtain a sufficient condition for their ergodicity, we will transform these balanced networks into a purely inhibitory network by "cutting" the excitatory connections, which amounts replacing all the original connectivity weights $w_{ij}$ by $\min(w_{ij}, 0)$. The countdown process of the original network is therefore upperbounded in law by the countdown process of the new process where we cut the inhibitory interactions. If the new process is ergodic, it implies that the original process is also. Therefore, we will be interested in proving ergodicity for purely inhibitory networks. As discussed in section 4, these networks can always be expressed as a countdown process with additional variables. Therefore we will be able to use the results obtained previously to get results on the new network.

## 5.1 Ergodicity of the PIF models

For the perfect integrate-and fire neuron, we have seen that the interaction variables $\eta_{ij}$ that we have to add only depend on $w_{ij}$ possibly weighted by a function depending on the last spike time of the postsynaptic cell $j$ in the case where we take into account the refractory period, and that may be added to the presynaptic neuron in the case where we take into account transmission delays. Note eventually that the ergodicity of the network is not influenced by the delays if taken into account.

Nevertheless in this case, both the expected value of the reset variable and of the interaction variable are infinite whatever the parameters. Indeed, they are expressed as first hitting times of Brownian motion and it is known that the expected values of these random

variables are infinite (see e.g. [23]). Therefore, it does not fit in the framework previously used. Nevertheless, in the simulations we did in this type of network, we obtain the same result as in the theoretical cases treated: for small synaptic connectivities, the network is recurrent, and each neuron will spike after any given time, but when the synaptic strength are too big, some neurons stop firing. In the case of a fully connected network, asymptotically only one neuron spikes after a given time, and in a linear or a ring network, one upon two neurons stops firing (see figure 10). The same type of behavior can therefore be observed as in the other cases, but still has to be mathematically characterized.

### 5.2 Ergodicity of the LIF models

The cases of the leaky integrate-and-fire models does not either fit in the previous framework developped, since the interaction random variable depends on the value of the countdown process at the time when it receives a spike. Nevertheless, both the reset and the interaction random variables have an expected value and are $\mathbb{L}^2$. The interaction variable is bounded by the variable associated with a countdown value equal to 0. Denote by $E_{ij}(x)$ the expected value of the interaction variable, where $x$ is the value of the countdown process when the interaction occurs. . In the case where we have $E(0)|\mathcal{V}(i)| < \mathbb{E}Y_i$ for all $i$, the network will be ergodic.

If this is not the case, then we can prove that when $x \to \infty$, we have $E(x) \to 0$ (we even prove that the interaction variable tends to 0 in law, see [23]). Therefore, the countdown value of the neuron will not tend to infinity in this case and always returns in the zone where $E(x)|\mathcal{V}(i)| \geq \mathbb{E}Y_i$. We conjecture that in this case, there is another condition on the synaptic weights for the network to be ergodic or not. Simulation results confort us in this conjecture

## 6 Numerical Simulations

As reviewed by Romain Brette and collaborators in [4], there are two main families of algorithms for the simulation of neural networks: synchronous or "clock-driven" algorithms, in which all neurons are updated simultaneously at every tick of a clock, and asynchronous or "event-driven" algorithms, in which neurons are updated only when they receive or emit a spike. We describe the simple clock-driven strategy to simulate this kind of neural network in section 6.1 and then we will study more precisely the implications of the above analysis to elaborate an event-driven simulator for stochastic networks in section 6.2

### 6.1 Clock-Driven simulation

In the synchronous or "clock-driven" algorithms, the state variables of all neurons are updated simultaneously at every tick of a clock ($X(t) \to X(t+dt)$, see figure 9), using a numerical integration algorithm. Then, after updating all the variables, the spiking condition is checked for every neuron. Each neuron that satisfies this condition produces a spike

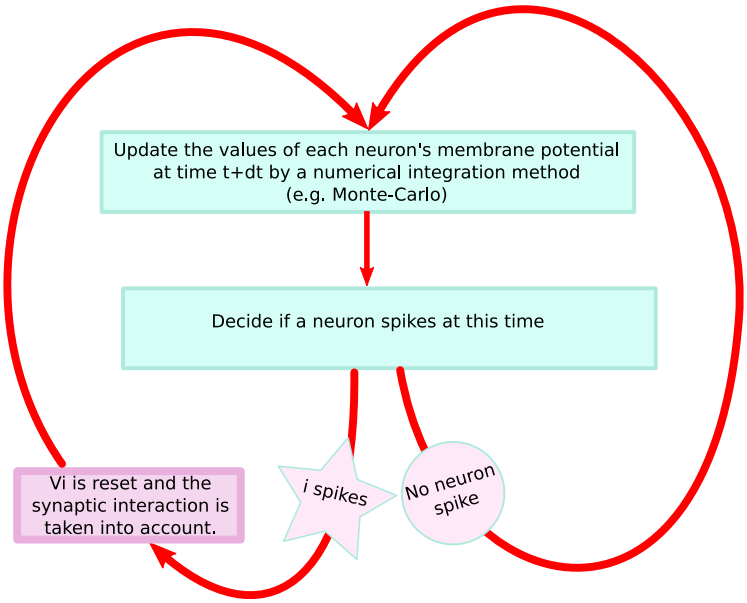



Figure 9: Monte-Carlo algorithm for the simulation of the stochastic neural network. The voltage potential is simulated at each time step and a decision is taken wether a spike is emitted or not.

which is transmitted with or without delay, and updates their corresponding variables. The membrane potential of every spiking cell is reset.

As reviewed in [4], the cost of the update phase is of order $N$ for each time step. For simulating the network during a time $T$, the complexity will therefore be $O(N\,T/dt)$. If $F$ is the average firing rate, an average of $F \times N$ spikes are produced by the neurons and each of these needs to be propagated to $p$ target neurons. Thus, the propagation phase consists in $F \times N \times p$ spike propagations per second. These are essentially additions of weights wi to state variables, and thus are simple operations whose cost does not grow with the complexity of the models. Summing up, the total mean computational cost per second of biological time is of order $O(N/dt + F\,N\,p)$. The cost of taking into account delays is not very high, and does not change the complexity of the algorithm. The obvious drawback of this type of algorithm is that spikes are aligned to a grid (ticks of the clock) thus the simulation is approximate even when the membrane potential is solved exactly. Furthermore, the spiking condition itself is checked at given times and therefore spikes can be missed. Many solutions to fix these issues have been proposed but none is really fully satisfactory.

For simulating a stochastic network with a synchronous algorithm, I used the Brian software [15], for its efficiency to deal with linear models. Indeed, computations are done

using matrix calculus and therefore are quite efficient. This simulation software based on python was not evaluated in the review [4] since it was produced after but we believe it is a good simulator for this type of linear equations. The code we used for perfect integrate-and-fire neuron is now freely accessible by downloading in the examples provided with the software.

The simulation results in this case are compatible with the results mathematically obtained in the previous study through the use of the Hourglass model: for small inhibitory connectivities, the ring network is ergodic and when the absolute value of the connectivities is big, one upon two neurons stop firing. Similarly we have been able to simulate the fully connected network. Results are given in figure 10. Note that the fact that spikes are aligned on a grid hide the ergocity on LIF networks, as illustrated in figure 11.

## 6.2 Event-driven simulation

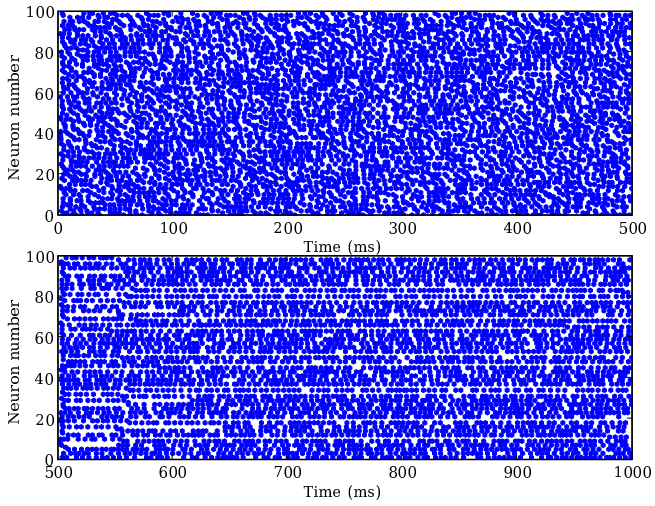
Another family of simulation strategies exist for neural network. This type of simulation is called asynchronous or "event-driven" algorithms. In that case neurons are updated only when they receive or emit a spike. This type of algorithm is less widely used than the clock-driven ones, because they are significantly more complex to implement and less universal. But the key advantage is the gain in speed and the fact that spike timings can be computed exactly (when possible, or can be approximated).

The approach developped in the previous sections provides a very natural way to define an event-based simulation algorithm for stochastic networks. This method consist in building a Markov chain describing the time of the spikes for each neuron. We have seen that simulating the times of the spikes is equivalent in law to simulating the membrane potential, from the spikes viewpoint. The event-based simulation consists in building this Markov chain. To this purpose, we have seen that we have to draw at each spike time in the law of two types of random variables: the reset random variable and the interaction random variables. These random variables can be expressed in most cases using the law of first hitting times of random processes. We studied the problem of describing the law of first hitting times of stochastic processes in [24, 25]. In the cases where these laws are known, a very efficient simulation procedure can be used. If they are not known in a closed form, then we will have to evaluate these random variables. We describe those two simulation ways in the following paragraphs.

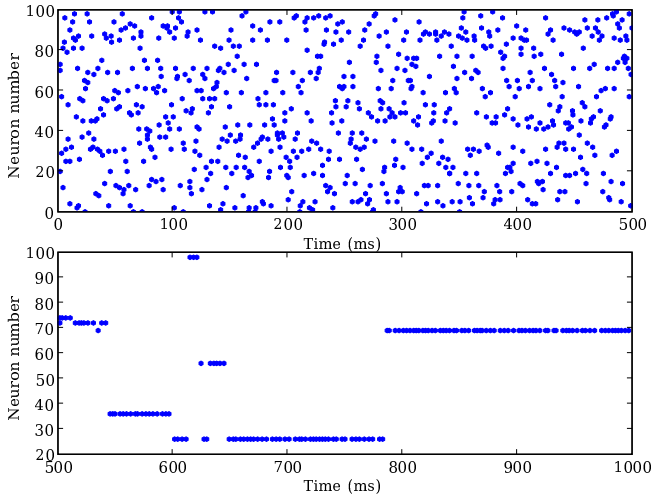
### 6.2.1 Known interaction variables

We consider a network of stochastic integrate-and-fire neurons such that the reset and the interaction variables are known, either analytically, or that computed offline using the techniques of [24, 25] and tabulated. In that case, simulating the related countdown process will be very efficient and will precisely give the spike times.

To define our event-based algorithm, we explain how to initialize the network and how to compute the spike times recursively. Assume that at the initial time $t_0$ the membrane potential of each neuron and of additional variables of the model are known. The initial




(a) Ring network of PIF




(b) Fully connected network of PIF

Figure 10: Clock-driven simulation of a 100 stochastic integrate-and-fire neurons network during $500ms$ using Brian software. In figure (a) we represented the ring network for small inhibitory connectivities and big inhibitory connectivities. We observe that one upon two neurons permanently stop firing. Figure (b) gives the result of the same type of simulations with a fully connected network. We observe that all but one neuron permanently stop firing.

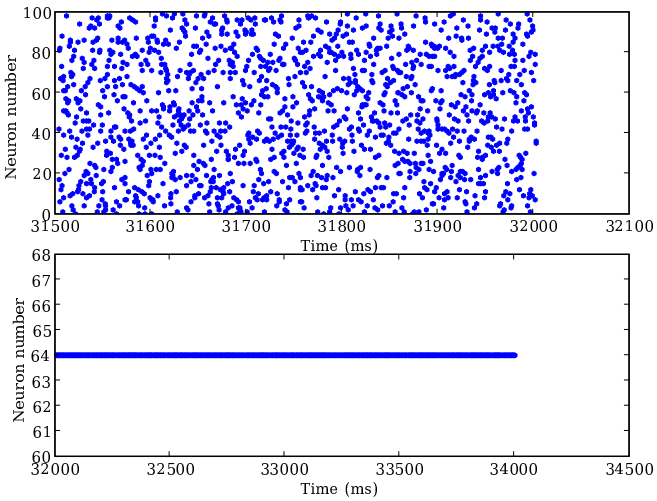



Figure 11: LIF network simulated with Brian software

countdown value for a given neuron will be simply computed as the first hitting time of its membrane potential process starting from this initial condition to reach the threshold and therefore can be computed in the same way as the reset variable. From this initial time, the principle of the algorithm is to build the discrete-time Markov chain containing as a variable the countdown process that gives the times of the spikes (we have seen that sometimes this variable needs additional variables to be simulated autonomously). Then to deduce the state of the chain at time $n+1$ knowing the chain at time $n$, we use the recursion relation described in sections 3 and 4 (see figure 12):

- We first identify the neuron having the lowest countdown value, which amounts finding the minimal value in a list of $N$ elements, which is very simple and efficient to code. This neuron is the one that elicits the first spike.
- When this neuron is identified, we directly jump to this time, and draw the new state of the network: the neuron that just fired a spike is reset by drawing in the law of the related reset variable and the other neurons' state is updated by drawing in the law of their respective interaction variables. Once the state of all neurons have been updated, the simulation proceeds.

This method was implemented using the software MVASpike [22, 21, 31, 4]. Mvaspike consists of a core C++ library, implementing a few generic classes to describe networks, neurons and additional input/output systems. It has been designed to be easy to access

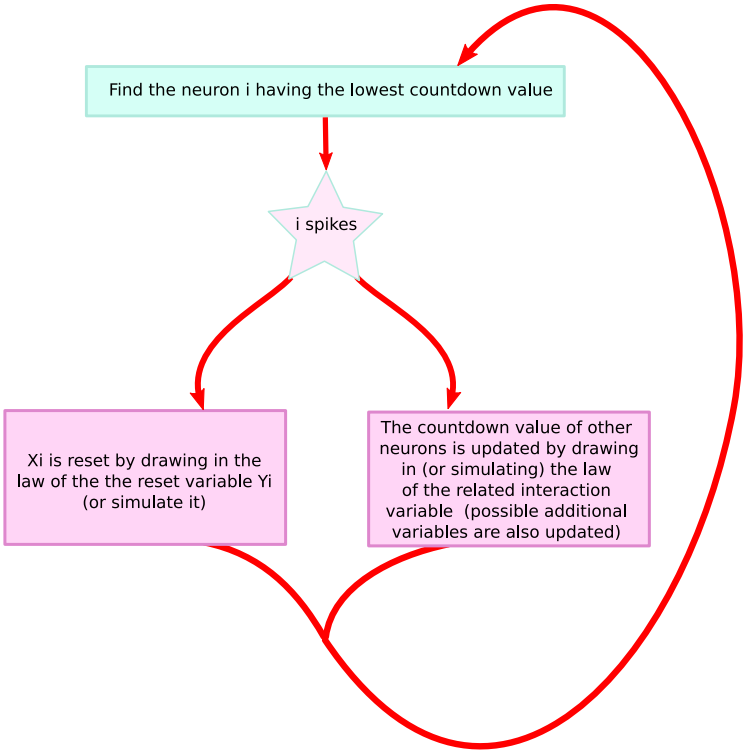



Figure 12: Principle of the event-based simulation using the countdown process studied in the previous sections

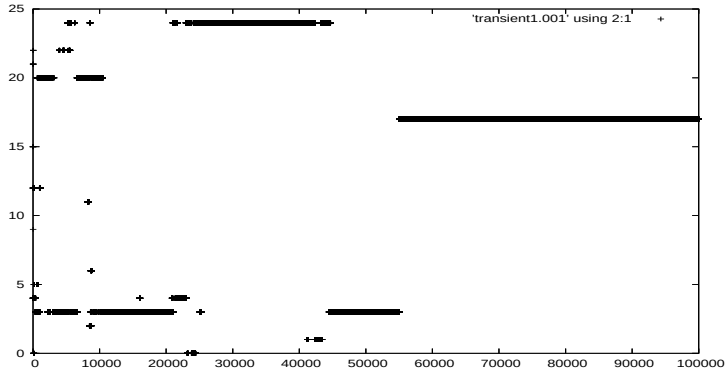
from other programming languages (high level or scripting languages, e.g. Python) and extensible. Well established simulation algorithms are provided, based on state of the art priority queue data structures. We did the same simulations as in the case of the clock-driven simulation.

To simulate with MVASpike networks of neuron whose pdf of the first hitting time is neither known analytical nor tabulated is quite unefficient since we have to compute it at each time step. Computing these variables amounts computing a trajectory of the membrane potential process, and therefore will be slower than the direct clock-based simulation. This issue can nevertheless be overcome by parallelizing these calculations.

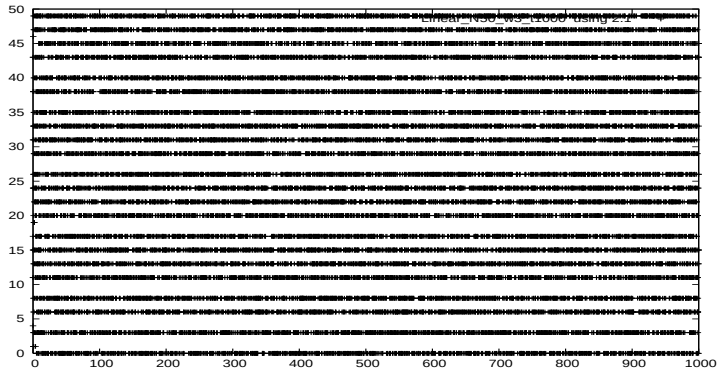
### 6.2.2 Parallel implementation

In the Monte-Carlo simulation of the hourglass model, we simulate the countdown processes computing at each time step the random variables of reset and interaction using a Monte-Carlo algorithm. This simulation is the only one available in the cases where the probability density functions of the first hitting time are neither analytically nor numerically known. It is the case for instance for complex models where the pdf cannot be tabulated offline, for instance when the input current is non stationary and depends on time. This type of simulation needs the user to define a clock, i.e. a time step for the simulation of trajectories in order to compute first hitting times of stochastic processes using a Monte-Carlo simulation. As a conclusion of the theoretical analysis driven above, an important remark is that at each spike time, the reset variables and the interaction variables are pairwise independent. Therefore they can be computed independently, and for instance at the same time using a parallel algorithm.

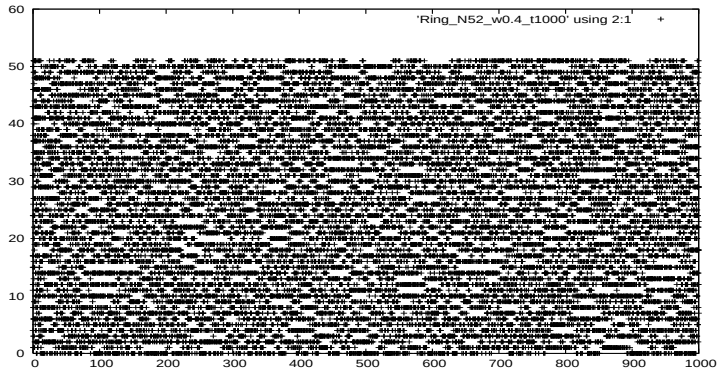
Therefore the idea was to implement the network on graphics processing unit (GPU), dedicated graphics rendering devices for personal computer. Modern GPUs are very efficient at manipulating and displaying computer graphics, and their highly parallel structure makes them more effective than general-purpose CPUs for a range of complex algorithms. Thanks to their high performance and programmability, the latest graphics cards can now be used for scientific purpose. They are indeed very efficient parallel Single Instruction Multiple Data (SIMD) machines. With the help of Renaud Keriven and Alexandre Chariot, we implemented the computation of the reset and interaction random variables on a GPU. One of the main issue of this problem was to build a random number generator. Indeed, usual graphical cards were not using integers. Very recent cards, starting from the cards NVidia 8xxx, are able to handle integers, and therefore it opened the way random number generators and random simulation. Another issue is the decorrelation between the random number generators on each processor. To this purpose, we generate random seeds on CPU to be used by the random number generation algorithm on each processor. After this common phase, each processor will behave independently. The processor that computes the reset variable by using a Monte-Carlo algorithm. This simulation is based on a pathwise simulation of the membrane potential and the evaluation of the spiking probability between two times steps. When at a given processor the random variable has been evaluated, the




(a) Fully connected network




(b) Linear network




(c) Recurrent network

Figure 13: Linear and fully connected networks of perfect integrate-and-fire neuron with constant inputs, simulated with MVASpike. Simulations of 25 to 50 neurons for $10s$ to $1000s$. We obtain the same results as expected from the mathematical analysis.

neuron is flagged, the simulation on this computer stops, the value of the random variable is recorded, and we wait for all the processor to reach this phase.

This process can be done for a number of neuron lower or equal to $4096^2$ ($\sim 1.6\,10^7$ units) because of the limited memory available on these cards. Nevertheless, we can overcome this difficulty by repeating many times this procedure.

With this algorithm we obtain spead up ratios from 20 to up to 100, by comparing with the same algorithm coded in C++.

## Conclusion

In this paper we developed an event-based mathematical framework for the study of stochastic integrate-and-fire neural networks. This model can be studied efficiently using the powerful tools of communication networks theory. With this approach we get ergodicity properties for the network, characterize the invariant measures, and can address biological questions. In contrast with other methods, no assumption has to be done on the connectivity map, on the number of connections or on the number of neuron, so this model can be used for instance to study cortical columns.

This study opens the door to the mathematical study of the macroscopic behaviour of large networks using the hydrodynamics limits developed to study large queuing processes, to infer and model collective behaviours of such networks.

## References

[1] Arbib. *The Handbook of Brain Theory and Neural Networks*. MIT Press, 1998.

[2] Soren Asmussen and Tatyana S. Turova. Stationarity properties of neural networks. *Journal of Applied Probabilities*, 35:783–794, 1998.

[3] A.A. Borovkov. *Ergodicity and stability of stochastic processes*. Wiley, Chichester, 1998.

[4] R. Brette, M. Rudolph, T. Carnevale, M. Hines, D. Beeman, J. M. Bower, M. Diesmann, A. Morrison, P. H. Goodman, F. C. Harris Jr., M. Zirpe, T. Natschläger, D. Pecevski, B. Ermentrout, M. Djurfeldt, A. Lansner, O. Rochel, T. Vieville, E. Muller, A. P. Davison, S. El Boustani, and A. Destexhe. Simulation of networks of spiking neurons: a review of tools and strategies. *Journal of Computational Neuroscience*, 23(3):349–398, 2007.

[5] N. Brunel and V. Hakim. Fast global oscillations in networks of integrate-and-fire neurons with low firing rates. *Neural Computation*, 11:1621–1671, 1999.

[6] M. Cottrell. Mathematical analysis of a neural network with inhibitory coupling. *Stochastic Processes and their Applications*, 40:103–127, 1992.

[7] M. Cottrell and T.S. Turova. Use of an hourglass model in neuronal coding. *Journal of applied probability*, 37:168–186, 2000.

[8] J. G. Dai. On positive harris recurrence of multiclass queueing networks: A unified approach via fluid limit models. *The Annals of Applied Probability*, 5:49–77, 1995.

[9] J. G. Dai. A fluid limit model criterion for instability of multiclass queueing networks. *Annals of Applied Probability*, 6:751–757, 1996.

[10] JG Dai and SP Meyer. Stability and convergence of moments for multiclass queuing networks via fluid limits models. *IEEE trans. on automatic control*, 40:1889–1904, 1995.

[11] M.H.A. Davis. Piecewise-deterministic markov processes: A general class of non-diffusion stochastic models. *Journal of the Royal Society, Series B (Methodological)*, 46:353–388, 1984.

[12] G. Fayolle, V.A. Malyshev, and MV Menshikov. *Topics in the Constructive Theory of Countable Markov Chains*. Cambridge University Press, 1995.

[13] Christine Fricker, Philippe Robert, Ellen Saada, and Danielle Tibi. Analysis of some networks with interaction. *Annals of Applied Probability*, 4:1112–1128, 1994.

[14] W. Gerstner and W. Kistler. *Spiking Neuron Models*. Cambridge University Press, 2002.

[15] D Goodman and R Brette. Brian: a simulator for spiking neural networks in python. *Frontiers in Neuroinformatics (in preparation)*, 2008.

[16] E.R. Kandel, J.H. Schwartz, and T.M. Jessel. *Principles of Neural Science*. McGraw-Hill, 4th edition, 2000.

[17] G. Last and H. Stamer. Recurrence and transience properties of some neural networks: an approach via fluid limit models. *Queueing Systems*, 32:99–130, 1999.

[18] VA Malyshev. Networks and dynamical systems. *Advances in applied probability*, 25(1):140–175, 1993.

[19] V.A. Malyshev and M.V. Menshikov. Ergodicity, continuity, and analyticity of countable markov chains. *Trans. Moscow Math. Soc.*, 1:1–47, 1981.

[20] Jan Reutimann, Michele Giugliano, and Stefano Fusi. Event-driven simulation of spiking neurons with stochastic dynamics. *Neural Computation*, 15:811–830, 2003.

[21] O. Rochel. *Une approche événementielle pour la modélisation et la simulation de neurones impulsionnels*. PhD thesis, Université Henri Poincaré - Nancy 1, 2004.

[22] O. Rochel and D. Martinez. An event-driven framework for the simulation of networks of spiking neurons. In *Proc. 11th European Symposium on Artificial Neural Networks*, pages 295–300, 2003.

[23] Jonathan Touboul. Stochastic processes and hitting times in mathematical neuroscience. Master's thesis, Universit'e Pierre et Marie Curie, 2006.

[24] Jonathan Touboul and Olivier Faugeras. The spikes trains probability distributions: a stochastic calculus approach. *Journal of Physiology, Paris*, 2007. To appear.

[25] Jonathan Touboul and Olivier Faugeras. First hitting time of double integral processes to curved boundaries. *Advances in Applied Probability*, 40(2):501–528, 2008.

[26] T. Turova. Stochastic dynamics of a neural network with inhibitory and excitatory connections. *BioSystems*, 40(1-2):197–202, 1997.

[27] Tatiana S. Turova. Analysis of a biological plausible neural network via an hourglass model. *Markov Processes and related fields*, 2:487–510, 1996.

[28] Tatyana Turova. Neural networks through the hourglass. *BioSystems*, 58:159–165, 2000.

[29] TS Turova, W. Mommaerts, and EC Van Der Meulen. Synchronization of firing times in a stochastic neural network model with excitatory connections. *Stochastic processes and their applications*, 50(1):173–186, 1994.

[30] A. van Rotterdam, F.H. Lopes da Silva, J. van den Ende, M.A. Viergever, and A.J. Hermans. A model of the spatial-temporal characteristics of the alpha rhythm. *Bulletin of Mathematical Biology*, 44(2):283–305, 1982.

[31] T. Viéville and O. Rochel. One step towards an abstract view of computation in spiking neural-networks. In *International Conf. on Cognitive and Neural Systems*, 2006.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399